(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) **EP 4 492 783 A1**

(12) **EUROPEAN PATENT APPLICATION**
published in accordance with Art. 153(4) EPC

(43) Date of publication:
15.01.2025 Bulletin 2025/03

(21) Application number: **23767115.1**

(22) Date of filing: **07.03.2023**

(51) International Patent Classification (IPC):
H04N 19/136 *(2014.01)* ***H04N 19/70*** *(2014.01)*
H04N 19/60 *(2014.01)* ***H04N 19/44*** *(2014.01)*
H04N 19/119 *(2014.01)* ***H04N 19/186*** *(2014.01)*
H04N 19/96 *(2014.01)*

(52) Cooperative Patent Classification (CPC):
**H04N 19/119; H04N 19/136; H04N 19/186;
H04N 19/44; H04N 19/60; H04N 19/70; H04N 19/96**

(86) International application number:
PCT/KR2023/003070

(87) International publication number:
WO 2023/172014 (14.09.2023 Gazette 2023/37)

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC ME MK MT NL NO PL PT RO RS SE SI SK SM TR
Designated Extension States:
BA
Designated Validation States:
KH MA MD TN

(30) Priority: **07.03.2022 KR 20220028818**

(71) Applicant: **LG Electronics Inc.
Yeongdeungpo-gu
Seoul 07336 (KR)**

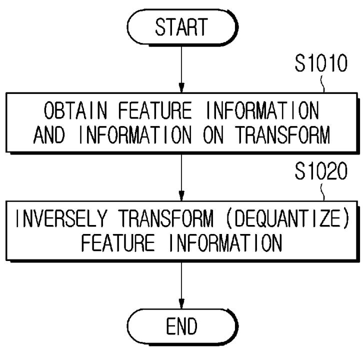
(72) Inventors:
- **KIM, Chulkeun
Seoul 06772 (KR)**
- **LIM, Jaehyun
Seoul 06772 (KR)**
- **GWAK, Donggyu
Seoul 06772 (KR)**

(74) Representative: **Maikowski & Ninnemann
Patentanwälte Partnerschaft mbB
Postfach 15 09 20
10671 Berlin (DE)**

(54) **FEATURE ENCODING/DECODING METHOD AND APPARATUS, AND RECORDING MEDIUM STORING BITSTREAM**

(57) Provided are a feature encoding/decoding method and device and a computer-readable recording medium generated by the feature encoding method. The feature decoding method according to the present disclosure, which is performed by the feature decoding device, comprises the steps of: obtaining information on transform of feature information and the feature information from a bitstream; and inversely transforming the feature information based on the information on the transform, wherein the information on the transform comprises information on a transform type that is used to transform the feature information.

FIG. 10



EP 4 492 783 A1

Description

Technical Field

[0001] The present disclosure relates to a feature encoding/decoding method and apparatus, and more specifically, to a feature encoding/decoding method and apparatus enabling development and application of an independent feature encoding apparatus and an independent feature decoding apparatus through abstraction of a feature extraction network, and a recording medium storing a bitstream generated by the feature encoding method/apparatus of the present disclosure.

Background Art

[0002] With the development of machine learning technology, demand for image processing-based artificial intelligence services is increasing. In order to effectively process a vast amount of image data required for artificial intelligence services within limited resources, image compression technology optimized for machine task performance is essential. However, existing image compression technology has been developed with the goal of high-resolution, high-quality image processing for human vision, and has the problem of being unsuitable for artificial intelligence services. Accordingly, research and development on new machine-oriented image compression technology suitable for artificial intelligence services is actively underway.

Disclosure

Technical Problem

[0003] An object of the present disclosure is to provide a feature encoding/decoding method and apparatus with improved encoding/decoding efficiency.

[0004] Another object of the present disclosure is to provide an independent feature encoding/decoding method and apparatus through abstraction of a feature extraction network.

[0005] Another object of the present disclosure is to provide a feature encoding/decoding method and apparatus capable of responding a change of a feature extraction network.

[0006] Another object of the present disclosure is to provide an independent feature encoding/decoding method and apparatus that transmit and obtain information on abstraction of a feature extraction network.

[0007] Another object of the present disclosure is to provide a method of transmitting a bitstream generated by a feature encoding method or apparatus according to the present disclosure.

[0008] Another object of the present disclosure is to provide a recording medium storing a bitstream generated by a feature encoding method or apparatus according to the present disclosure.

[0009] Another object of the present disclosure is to provide a recording medium storing a bitstream received, decoded and used to reconstruct a feature by a feature decoding apparatus according to the present disclosure.

[0010] The technical problems solved by the present disclosure are not limited to the above technical problems and other technical problems which are not described herein will become apparent to those skilled in the art from the following description.

Technical Solution

[0011] A feature decoding method according to an aspect of the present disclosure, which is performed by a feature decoding apparatus, may comprise obtaining information on transform of feature information and the feature information from a bitstream and inversely transforming the feature information based on the information on the transform, and the information on the transform may comprise information on a transform type that is used to transform the feature information.

[0012] A feature encoding method according to another aspect of the present disclosure, which is performed by a feature encoding apparatus, may comprise obtaining feature information of input data from the input data, generating transformed feature information by transforming the feature information, and encoding the transformed feature information and information on transform, and the information on the transform may comprise information on a transform type that is used to transform the feature information.

[0013] A recording medium according to another aspect of the present disclosure may store a bitstream generated by the feature encoding method or the feature encoding apparatus of the present disclosure.

[0014] A bitstream transmission method according to another aspect of the present disclosure may transmit a bitstream generated by the feature encoding method or the feature encoding apparatus of the present disclosure to a feature decoding apparatus.

[0015] The features briefly summarized above with respect to the present disclosure are merely exemplary aspects of the detailed description below of the present disclosure, and do not limit the scope of the present disclosure.

Advantageous Effects

[0016] According to the present disclosure, it is possible to provide a feature information encoding/decoding method and apparatus with improved encoding/decoding efficiency.

[0017] Also, according to the present disclosure, it is possible to provide a feature encoding/decoding method and apparatus capable of adaptively responding a change of a feature extraction network.

[0018] Also, according to the present disclosure, it is possible to guarantee a normal operation of a feature encoding/decoding method and apparatus by adaptively responding a change of a feature extraction network.

[0019] Also, according to the present disclosure, it is possible to satisfy a requirement of VCM that VCM should operate independently of a network model and a machine task.

[0020] It will be appreciated by persons skilled in the art that that the effects that can be achieved through the present disclosure are not limited to what has been particularly described hereinabove and other advantages of the present disclosure will be more clearly understood from the detailed description.

Description of Drawings

[0021]

FIG. 1 is a view schematically showing a VCM system to which embodiments of the present disclosure are applicable.
FIG. 2 is a diagram schematically showing a VCM pipeline structure to which embodiments of the present disclosure are applicable.
FIG. 3 is a diagram schematically showing an image/video encoder to which embodiments of the present disclosure are applicable.
FIG. 4 is a diagram schematically showing an image/video decoder to which embodiments of the present disclosure are applicable.
FIG. 5 is a flowchart schematically illustrating a feature/feature map encoding procedure to which embodiments of the present disclosure are applicable.
FIG. 6 is a flowchart schematically illustrating a feature/feature map decoding procedure to which embodiments of the present disclosure are applicable.
FIG. 7 and FIG. 8 are views showing examples of feature encoding and decoding structures to which embodiments of the present disclosure are applicable.
FIG. 9 is a flowchart showing a feature encoding method according to an embodiment of the present disclosure.
FIG. 10 is a flowchart showing a feature decoding method according to an embodiment of the present disclosure.
FIG. 11 is a flowchart showing a feature encoding method according to another embodiment of the present disclosure.
FIG. 12 is a flowchart showing a feature decoding method according to another embodiment of the present disclosure.
FIG. 13 is a view for explaining an example of determining a transform type and an inverse transform type.
FIG. 14 is a flowchart showing a feature encoding method according to yet another embodiment of the present disclosure.
FIG. 15 is a flowchart showing a feature decoding method according to yet another embodiment of the present disclosure.
FIG. 16 is a flowchart showing an example of feature information and information on transform at feature encoding and decoding steps to which embodiments of the present disclosure are applicable.
FIG. 17 is a view showing an example of a content streaming system to which embodiments of the present disclosure are applicable.
FIG. 18 is a view showing another example of a content streaming system to which embodiments of the present disclosure are applicable.

Mode for Invention

[0022] Hereinafter, the embodiments of the present disclosure will be described in detail with reference to the accompanying drawings so as to be easily implemented by those skilled in the art. However, the present disclosure may be implemented in various different forms, and is not limited to the embodiments described herein.

[0023] In describing the present disclosure, in case it is determined that the detailed description of a related known function or construction renders the scope of the present disclosure unnecessarily ambiguous, the detailed description thereof will be omitted. In the drawings, parts not related to the description of the present disclosure are omitted, and similar reference numerals are attached to similar parts.

[0024] In the present disclosure, when a component is "connected", "coupled" or "linked" to another component, it may include not only a direct connection relationship but also an indirect connection relationship in which an intervening component is present. In addition, when a component "includes" or "has" other components, it means that other components may be further included, rather than excluding other components unless otherwise stated.

[0025] In the present disclosure, the terms first, second, etc. may be used only for the purpose of distinguishing one component from other components, and do not limit the order or importance of the components unless otherwise stated. Accordingly, within the scope of the present disclosure, a first component in one embodiment may be referred to as a second component in another embodiment, and similarly, a second component in one embodiment may be referred to as a first component in another embodiment.

[0026] In the present disclosure, components that are distinguished from each other are intended to clearly describe each feature, and do not mean that the components are necessarily separated. That is, a plurality of components may be integrated and implemented in one hardware or software unit, or one component may be distributed and implemented in a plurality of hardware or software units. Therefore, even if not stated otherwise, such embodiments in which the components are integrated or the component is distributed are also included in the scope of the present disclosure.

[0027] In the present disclosure, the components described in various embodiments do not necessarily mean essential components, and some components may be optional components. Accordingly, an embodiment consisting of a subset of components described in an embodiment is also included in the scope of the present disclosure. In addition, embodiments including other components in addition to components described in the various embodiments are included in the scope of the present disclosure.

[0028] The present disclosure relates to encoding and decoding of an image, and terms used in the present disclosure may have a general meaning commonly used in the technical field, to which the present disclosure belongs, unless newly defined in the present disclosure.

[0029] The present disclosure may be applied to a method disclosed in a Versatile Video Coding (VVC) standard and/or a Video Coding for Machines (VCM) standard. In addition, the present disclosure may be applied to a method disclosed in an essential video coding (EVC) standard, AOMedia Video 1 (AV1) standard, 2nd generation of audio video coding standard (AVS2), or a next-generation video/image coding standard (e.g., H.267 or H.268, etc.).

[0030] This disclosure provides various embodiments related to video/image coding, and, unless otherwise stated, the embodiments may be performed in combination with each other. In the present disclosure, "video" refers to a set of a series of images according to the passage of time. An "image" may be information generated by artificial intelligence (AI). Input information used in the process of performing a series of tasks by AI, information generated during the information processing process, and the output information may be used as images. In the present disclosure, a "picture" generally refers to a unit representing one image in a specific time period, and a slice/tile is a coding unit constituting a part of a picture in encoding. One picture may be composed of one or more slices/tiles. In addition, a slice/tile may include one or more coding tree units (CTUs). The CTU may be partitioned into one or more CUs. A tile is a rectangular region present in a specific tile row and a specific tile column in a picture, and may be composed of a plurality of CTUs. A tile column may be defined as a rectangular region of CTUs, may have the same height as a picture, and may have a width specified by a syntax element signaled from a bitstream part such as a picture parameter set. A tile row may be defined as a rectangular region of CTUs, may have the same width as a picture, and may have a height specified by a syntax element signaled from a bitstream part such as a picture parameter set. A tile scan is a certain continuous ordering method of CTUs partitioning a picture. Here, CTUs may be sequentially ordered according to a CTU raster scan within a tile, and tiles in a picture may be sequentially ordered according to a raster scan order of tiles of the picture. A slice may contain an integer number of complete tiles, or may contain a continuous integer number of complete CTU rows within one tile of one picture. A slice may be exclusively included in a single NAL unit. One picture may be composed of one or more tile groups. One tile group may include one or more tiles. A brick may indicate a rectangular region of CTU rows within a tile in a picture. One tile may include one or more bricks. The brick may refer to a rectangular region of CTU rows in a tile. One tile may be split into a plurality of bricks, and each brick may include one or more CTU rows belonging to a tile. A tile which is not split into a plurality of bricks may also be treated as a brick.

[0031] In the present disclosure, a "pixel" or a "pel" may mean a smallest unit constituting one picture (or image). In addition, "sample" may be used as a term corresponding to a pixel. A sample may generally represent a pixel or a value of a pixel, and may represent only a pixel/pixel value of a luma component or only a pixel/pixel value of a chroma component.

[0032] In an embodiment, especially when applied to VCM, when there is a picture composed of a set of components having different characteristics and meanings, a pixel/pixel value may represent a pixel/pixel value of a component generated through independent information or combination, synthesis, and analysis of each component. For example, in RGB input, only the pixel/pixel value of R may be represented, only the pixel/pixel value of G may be represented, or only the pixel/pixel value of B may be represented. For example, only the pixel/pixel value of a luma component synthesized using the R, G, and B components may be represented. For example, only the pixel/pixel values of images and information extracted through analysis of R, G, and B components from components may be represented.

[0033] In the present disclosure, a "unit" may represent a basic unit of image processing. The unit may include at least one of a specific region of the picture and information related to the region. One unit may include one luma block and two chroma (e.g., Cb and Cr) blocks. The unit may be used interchangeably with terms such as "sample array", "block" or "area" in some cases. In a general case, an M×N block may include samples (or sample arrays) or a set (or array) of transform coefficients of M columns and N rows. In an embodiment, In particular, especially when applied to VCM, the unit may represent a basic unit containing information for performing a specific task.

[0034] In the present disclosure, "current block" may mean one of "current coding block", "current coding unit", "coding target block", "decoding target block" or "processing target block". When prediction is performed, "current block" may mean "current prediction block" or "prediction target block". When transform (inverse transform)/quantization (dequantization) is performed, "current block" may mean "current transform block" or "transform target block". When filtering is performed, "current block" may mean "filtering target block".

[0035] In addition, in the present disclosure, a "current block" may mean "a luma block of a current block" unless explicitly stated as a chroma block. The "chroma block of the current block" may be expressed by including an explicit description of a chroma block, such as "chroma block" or "current chroma block".

[0036] In the present disclosure, the term "/" and "," should be interpreted to indicate "and/or." For instance, the expression "A/B" and "A, B" may mean "A and/or B." Further, "A/B/C" and "A/B/C" may mean "at least one of A, B, and/or C."

[0037] In the present disclosure, the term "or" should be interpreted to indicate "and/or." For instance, the expression "A or B" may comprise 1) only "A", 2) only "B", and/or 3) both "A and B". In other words, in the present disclosure, the term "or" should be interpreted to indicate "additionally or alternatively."

[0038] The present disclosure relates to video/image coding for machines (VCM).

[0039] VCM refers to a compression technology that encodes/decodes part of a source image/video or information obtained from the source image/video for the purpose of machine vision. In VCM, the encoding/decoding target may be referred to as a feature. The feature may refer to information extracted from the source image/video based on task purpose, requirements, surrounding environment, etc. The feature may have a different information form from the source image/video, and accordingly, the compression method and expression format of the feature may also be different from those of the video source.

[0040] VCM may be applied to a variety of application fields. For example, in a surveillance system that recognizes and tracks objects or people, VCM may be used to store or transmit object recognition information. In addition, in intelligent transportation or smart traffic systems, VCM may be used to transmit vehicle location information collected from GPS, sensing information collected from LIDAR, radar, etc., and various vehicle control information to other vehicles or infrastructure. Additionally, in the smart city field, VCM may be used to perform individual tasks of interconnected sensor nodes or devices.

[0041] The present disclosure provides various embodiments of feature/feature map coding. Unless otherwise specified, embodiments of the present disclosure may be implemented individually, or may be implemented in combination of two or more.

Overview of VCM System

[0042] FIG. 1 is a diagram schematically showing a VCM system to which embodiments of the present disclosure are applicable.

[0043] Referring to FIG. 1, the VCM system may include an encoding apparatus 10 and a decoding apparatus 20.

[0044] The encoding apparatus 10 may generate a bitstream by compressing/encoding a feature/feature map extracted from a source image/video and transmit the generated bitstream to the decoding apparatus 20 through a storage medium or a network. The encoding apparatus 10 may also be referred to as a feature encoding apparatus. In a VCM system, the feature/feature map may be generated at each hidden layer of a neural network. The size and number of channels of the generated feature map may vary depending on the type of neural network or the location of the hidden layer. In the present disclosure, a feature map may be referred to as a feature set, and a feature or a feature map may be referred to as 'feature information'.

[0045] The encoding apparatus 10 may include a feature acquisition unit 11, an encoding unit 12, and a transmission unit 13.

[0046] The feature acquisition unit 11 may acquire a feature/feature map for the source image/video. Depending on the embodiment, the feature acquisition unit 11 may acquire a feature/feature map from an external device, for example, a feature extraction network. In this case, the feature acquisition unit 11 performs a feature reception interface function. Alternatively, the feature acquisition unit 11 may acquire a feature/feature map by executing a neural network (e.g., CNN, DNN, etc.) using the source image/video as input. In this case, the feature acquisition unit 11 performs a feature extraction network function.

[0047] Depending on the embodiment, the encoding apparatus 10 may further include a source image generator (not shown) for acquiring the source image/video. The source image generator may be implemented with an image sensor, a camera module, etc., and may acquire the source image/video through an image/video capture, synthesis, or generation process. In this case, the generated source image/video may be sent to the feature extraction network and used as input data for extracting the feature/feature map.

[0048] The encoding unit 12 may encode the feature/feature map acquired by the feature acquisition unit 11. The encoding unit 12 may perform a series of procedures such as prediction, transform, and quantization to increase encoding efficiency. The encoded data (encoded feature/feature map information) may be output in the form of a bitstream. The bitstream containing the encoded feature/feature map information may be referred to as a VCM bitstream.

[0049] The transmission unit 13 may obtain feature/feature map information or data output in the form of a bitstream and forward the feature/feature map information or data to the decoding apparatus 20 or another external object through a digital storage medium or network in the form of a file or streaming. Herein, the digital storage medium may include various storage media such as USB, SD, CD, DVD, Blu-ray, HDD, and SSD. The transmission unit 13 may include elements for generating a media file with a predetermined file format or elements for transmitting data through a broadcasting/communication network. The transmission unit 13 may be provided as a separate transmission device from the encoding unit 12, and in this case the transmission device may include at least one processor for obtaining feature/feature map information or data output in the form of a bitstream and a transmission unit for forwarding the feature/feature map information or data in the form of a file or streaming.

[0050] The decoding apparatus 20 may acquire feature/feature map information from the encoding apparatus 10 and reconstruct the feature/feature map based on the acquired information.

[0051] The decoding apparatus 20 may include a reception unit 21 and a decoding unit 22.

[0052] The reception unit 21 may receive a bitstream from the encoding apparatus 10, acquire feature/feature map information from the received bitstream, and send it to the decoding unit 22.

[0053] The decoding unit 22 may decode the feature/feature map based on the acquired feature/feature map information. The decoding unit 22 may perform a series of procedures such as dequantization, inverse transform, and prediction corresponding to the operation of the encoding unit 12 to increase decoding efficiency.

[0054] Depending on the embodiment, the decoding apparatus 20 may further include a task analysis/rendering unit 23.

[0055] The task analysis/rendering unit 23 may perform task analysis based on the decoded feature/feature map. Additionally, the task analysis/rendering unit 23 may render the decoded feature/feature map into a form suitable for task performance. Various machine (oriented) tasks may be performed based on task analysis results and the rendered features/feature map.

[0056] As described above, the VCM system may encode/decode the feature extracted from the source image/video according to user and/or machine requests, task purpose, and surrounding environment, and performs various machine (oriented) tasks based on the decoded feature. The VCM system may be implemented by expanding/redesigning the video/image coding system and may perform various encoding/decoding methods defined in the VCM standard.

VCM pipeline

[0057] FIG. 2 is a diagram schematically showing a VCM pipeline structure to which embodiments of the present disclosure are applicable.

[0058] Referring to FIG. 2, the VCM pipeline 200 may include a first pipeline 210 for encoding/decoding an image/video and a second pipeline 220 for encoding/decoding a feature/feature map. In the present disclosure, the first pipeline 210 may be referred to as a video codec pipeline, and the second pipeline 220 may be referred to as a feature codec pipeline.

[0059] The first pipeline 210 may include a first stage 211 for encoding an input image/video and a second stage 212 for decoding the encoded image/video to generate a reconstructed image/video. The reconstructed image/video may be used for human viewing, that is, human vision.

[0060] The second pipeline 220 may include a third stage 221 for extracting a feature/feature map from the input image/video, a fourth stage 222 for encoding the extracted feature/feature map, and a fifth stage 223 for decoding the encoded feature/feature map to generate a reconstructed feature/feature map. The reconstructed feature/feature map may be used for a machine (vision) task. Here, the machine (vision) task may refer to a task in which images/videos are consumed by a machine. The machine (vision) task may be applied to service scenarios such as, for example, Surveillance, Intelligent Transportation, Smart City, Intelligent Industry, Intelligent Content, etc. Depending on the embodiment, the reconstructed feature/feature map may be used for human vision.

[0061] Depending on the embodiment, the feature/feature map encoded in the fourth stage 222 may be transferred to the first stage 221 and used to encode the image/video. In this case, an additional bitstream may be generated based on the encoded feature/feature map, and the generated additional bitstream may be transferred to the second stage 222 and used to decode the image/video.

[0062] Depending on the embodiment, the feature/feature map decoded in the fifth stage 223 may be transferred to the second stage 222 and used to decode the image/video.

[0063] FIG. 2 shows a case where the VCM pipeline 200 includes a first pipeline 210 and a second pipeline 220, but this is merely an example and embodiments of the present disclosure are not limited thereto. For example, the VCM pipeline 200 may include only the second pipeline 220, or the second pipeline 220 may be expanded into multiple feature codec pipelines.

[0064] Meanwhile, in the first pipeline 210, the first stage 211 may be performed by an image/video encoder, and the second stage 212 may be performed by an image/video decoder. Additionally, in the second pipeline 220, the third stage 221 may be performed by a VCM encoder (or feature/feature map encoder), and the fourth stage 222 may be performed by a VCM decoder (or feature/feature map encoder). Hereinafter, the encoder/decoder structure will be described in detail.

Encoder

[0065] FIG. 3 is a diagram schematically showing an image/video encoder to which embodiments of the present disclosure are applicable.

[0066] Referring to FIG. 3, the image/video encoder 300 may further include an image partitioner 310, a predictor 320, a residual processor 330, an entropy encoder 340, and an adder 350, a filter 360, and a memory 370. The predictor 320 may include an inter predictor 321 and an intra predictor 322. The residual processor 330 may include a transformer 332, a quantizer 333, a dequantizer 334, and an inverse transformer 335. The residual processor 330 may further include a subtractor 331. The adder 350 may be referred to as a reconstructor or a reconstructed block generator. The image partitioner 310, the predictor 320, the residual processor 330, the entropy encoder 340, the adder 350, and the filter 360 may be configured by one or more hardware components (e.g., encoder chipset or processor) depending on the embodiment. Additionally, the memory 370 may include a decoded picture buffer (DPB) and may be configured by a digital storage medium. The hardware components described above may further include a memory 370 as an internal/external component.

[0067] The image partitioner 310 may partition an input image (or picture, frame) input to the image/video encoder 300 into one or more processing units. As an example, the processing unit may be referred to as a coding unit (CU). The coding unit may be recursively partitioned according to a quad-tree binary-tree ternary-tree (QTBTTT) structure from a coding tree unit (CTU) or largest coding unit (LCU). For example, one coding unit may be partitioned into a plurality of coding units of deeper depth based on a quad tree structure, binary tree structure, and/or ternary structure. In this case, for example, the quad tree structure may be applied first and the binary tree structure and/or ternary structure may be applied later. Alternatively, the binary tree structure may be applied first. The image/video coding procedure according to the present disclosure may be performed based on a final coding unit that is no longer partitioned. In this case, the largest coding unit may be used as the final coding unit based on coding efficiency according to image characteristics, or, if necessary, the coding unit may be recursively partitioned into coding units of deeper depth to use a coding unit with an optimal size as the final coding unit. Here, the coding procedure may include procedures such as prediction, transform, and reconstruction, which will be described later. As another example, the processing unit may further include a prediction unit (PU) or a transform unit (TU). In this case, the prediction unit and the transform unit may each be divided or partitioned from the final coding unit described above. The prediction unit may be a unit of sample prediction, and the transform unit may be a unit for deriving a transform coefficient and/or a unit for deriving a residual signal from the transform coefficient.

[0068] In some cases, the unit may be used interchangeably with terms such as block or area. In a general case, an MxN block may represent a set of samples or transform coefficients consisting of M columns and N rows. A sample may generally represent a pixel or a pixel value, and may represent only a pixel/pixel value of a luma component, or only a pixel/pixel value of a chroma component. The sample may be used as a term corresponding to pixel or pel.

[0069] The image/video encoder 300 may generate a residual signal (residual block, residual sample array) by subtracting a prediction signal (predicted block, prediction sample array) output from the inter predictor 321 or the intra predictor 322 from the input image signal (original block, original sample array) and transmit the generated residual signal to the transformer 332. In this case, as shown, the unit that subtracts the prediction signal (prediction block, prediction sample array) from the input image signal (original block, original sample array) within the image/video encoder 300 may be referred to as the subtractor 331. The predictor may perform prediction on a processing target block (hereinafter referred to as a current block) and generate a predicted block including prediction samples for the current block. The predictor may determine whether intra prediction or inter prediction is applied in current block or CU units. The predictor may generate various information related to prediction, such as prediction mode information, and transfer it to the entropy encoder 340. Information about prediction may be encoded in the entropy encoder 340 and output in the form of a bitstream.

[0070] The intra predictor 322 may predict the current block by referring to the samples in the current picture. At this time, the referenced samples may be located in the neighbor of the current block or may be located away from the current block, depending on the prediction mode. In intra prediction, the prediction modes may include a plurality of non-directional modes and a plurality of directional modes. The non-directional mode may include, for example, a DC mode and a planar mode. The directional mode may include, for example, 33 directional prediction modes or 65 directional prediction modes according to the degree of detail of the prediction direction. However, this is merely an example, more or less directional prediction modes may be used depending on settings. The intra predictor 322 may determine the prediction mode applied to the current block by using a prediction mode applied to a neighboring block.

[0071] The inter predictor 321 may derive a predicted block for the current block based on a reference block (reference sample array) specified by a motion vector on a reference picture. In this case, in order to reduce the amount of motion information transmitted in the inter prediction mode, the motion information may be predicted in block, subblock, or sample units based on correlation of motion information between the neighboring block and the current block. The motion information may include a motion vector and a reference picture index. The motion information may further include inter prediction direction (L0 prediction, L1 prediction, Bi prediction, etc.) information. In the case of inter prediction, the neighboring block may include a spatial neighboring block present in the current picture and a temporal neighboring block present in the reference picture. The reference picture including the reference block and the reference picture including the temporal neighboring block may be the same or different. The temporal neighboring block may be called a collocated reference block, a co-located CU (colCU), and the like, and the reference picture including the temporal neighboring block may be called a collocated picture (colPic). For example, the inter predictor 321 may construct a motion information candidate list based on neighboring blocks and generate information indicating which candidate is used to derive a motion vector and/or reference picture index of the current block. Inter prediction may be performed based on various prediction modes, and, for example, in the case of a skip mode and a merge mode, the inter predictor 321may use motion information of the neighboring block as motion information of the current block. In the case of the skip mode, unlike the merge mode, the residual signal may not be transmitted. In the case of the motion vector prediction (MVP) mode, the motion vector of the neighboring block may be used as a motion vector predictor, and a motion vector difference may be signaled to indicate the motion vector of the current block.

[0072] The predictor 320 may generate a prediction signal based on various prediction methods. For example, the predictor may not only apply intra prediction or inter prediction but also simultaneously apply both intra prediction and inter prediction, for prediction of one block. This may be called combined inter and intra prediction (CIIP). In addition, the predictor may be based on an intra block copy (IBC) prediction mode or a palette mode for prediction of the block. The IBC prediction mode or the palette mode may be used for content image/video coding of a game or the like, for example, screen content coding (SCC). IBC basically performs prediction within the current picture, but may be performed similarly to inter prediction in that a reference block is derived within the current picture. That is, IBC may use at least one of the inter prediction techniques described in the present disclosure. The palette mode may be regarded as an example of intra coding or intra prediction. When the palette mode is applied, the sample values within the picture may be signaled based on information about a palette table and a palette index.

[0073] The prediction signal generated by the predictor 320 may be used to generate a reconstructed signal or to generate a residual signal. The transformer 332 may generate transform coefficients by applying a transform technique to the residual signal. For example, the transform technique may include at least one of a discrete cosine transform (DCT), a discrete sine transform (DST), a karhunen-loève transform (KLT), a graph-based transform (GBT), or a conditionally non-linear transform (CNT). Here, the GBT refers to transform obtained from a graph when relationship information between pixels is represented by the graph. The CNT refers to transform acquired based on a prediction signal generated using all previously reconstructed pixels. In addition, the transform process may be applied to square pixel blocks having the same size or may be applied to non-square blocks having a variable size.

[0074] The quantizer 130 may quantize the transform coefficients and transmit them to the entropy encoder 190. The entropy encoder 190 may encode the quantized signal (information on the quantized transform coefficients) and output a bitstream. The information on the quantized transform coefficients may be referred to as residual information. The quantizer 130 may reorder quantized transform coefficients in a block form into a one-dimensional vector form based on a coefficient scan order and generate information on the quantized transform coefficients based on the quantized transform coefficients in the one-dimensional vector form. The entropy encoder 340 may perform various encoding methods such as, for example, exponential Golomb, context-adaptive variable length coding (CAVLC), context-adaptive binary arithmetic coding (CABAC), and the like. The entropy encoder 340 may encode information necessary for video/image reconstruction other than quantized transform coefficients (e.g., values of syntax elements, etc.) together or separately. Encoded information (e.g., encoded video/image information) may be transmitted or stored in units of network abstraction layers (NALs) in the form of a bitstream. The video/image information may further include information on various parameter sets such as an adaptation parameter set (APS), a picture parameter set (PPS), a sequence parameter set (SPS), or a video parameter set (VPS). In addition, the video/image information may further include general constraint information. In addition, the video/image information may further include a method of generating and using encoded information, a purpose, and the like. In the present disclosure, information and/or syntax elements transferred/signaled from the image/video encoder to the image/video decoder may be included in image/video information. The image/video information may be encoded through the above-described encoding procedure and included in the bitstream. The bitstream may be transmitted over a network or may be stored in a digital storage medium. The network may include a broadcasting network and/or a communication network, and the digital storage medium may include various storage media such as USB, SD, CD, DVD, Blu-ray, HDD, SSD, and the like. A transmitter (not shown) transmitting a signal output from the entropy encoder 340 and/or a storage unit (not shown) storing the signal may be configured as internal/external element of the image/video encoder 300, or the transmitter may be included in the entropy encoder 340.

[0075] The quantized transform coefficients output from the quantizer 130 may be used to generate a prediction signal. For example, the residual signal (residual block or residual samples) may be reconstructed by applying dequantization and inverse transform to the quantized transform coefficients through the dequantizer 334 and the inverse transformer 335. The adder 350 adds the reconstructed residual signal to the prediction signal output from the inter predictor 321 or the intra predictor 322 to generate a reconstructed signal (reconstructed picture, reconstructed block, reconstructed sample array). In a case where there is no residual for the processing target block, such as a case where the skip mode is applied, the predicted block may be used as the reconstructed block. The adder 350 may be called a reconstructor or a reconstructed block generator. The generated reconstructed signal may be used for intra prediction of a next processing target block in the current picture and may be used for inter prediction of a next picture through filtering as described below.

[0076] Meanwhile, luma mapping with chroma scaling (LMCS) is applicable in a picture encoding and/or reconstruction process.

[0077] The filter 360 may improve subjective/objective image quality by applying filtering to the reconstructed signal. For example, the filter 360 may generate a modified reconstructed picture by applying various filtering methods to the reconstructed picture, and store the modified reconstructed picture in the memory 370, specifically, a DPB of the memory 370. The various filtering methods may include, for example, deblocking filtering, a sample adaptive offset, an adaptive loop filter, a bilateral filter, and the like. The filter 360 may generate various information related to filtering and transmit the generated information to the entropy encoder 190. The information related to filtering may be encoded by the entropy encoder 340 and output in the form of a bitstream.

[0078] The modified reconstructed picture transmitted to the memory 370 may be used as the reference picture in the inter predictor 321. Through this, prediction mismatch between the encoder and the decoder may be avoided and encoding efficiency may be improved.

[0079] The DPB of the memory 370 may store the modified reconstructed picture for use as a reference picture in the inter predictor 321. The memory 370 may store the motion information of the block from which the motion information in the current picture is derived (or encoded) and/or the motion information of the blocks in the already reconstructed picture. The stored motion information may be transferred to the inter predictor 321 for use as the motion information of the spatial neighboring block or the motion information of the temporal neighboring block. The memory 370 may store reconstructed samples of reconstructed blocks in the current picture and may transfer the stored reconstructed samples to the intra predictor 322.

[0080] Meanwhile, the VCM encoder (or feature/feature map encoder) basically performs a series of procedures such as prediction, transform, and quantization to encode the feature/feature map and thus may basically have the same/similar structure as the image/video encoder 300 described with reference to FIG. 3. However, the VCM encoder is different from the image/video encoder 300 in that the feature/feature map is an encoding target, and thus may be different from the image/video encoder 300 in the name of each unit (or component) (e.g., image partitioner 310, etc.) and its specific operation content. The specific operation of the VCM encoder will be described in detail later.

Decoder

[0081] FIG. 4 is a diagram schematically showing an image/video decoder to which embodiments of the present disclosure are applicable.

[0082] Referring to FIG. 4, the image/video decoder 400 may include an entropy decoder 410, a residual processor 420, a predictor 430, an adder 440, a filter 450 and a memory 460. The predictor 430 may include an inter predictor 431 and an intra predictor 432. The residual processor 420 may include a dequantizer 421 and an inverse transformer 422. The entropy decoder 410, the residual processor 420, the predictor 430, the adder 440, and the filter 450 may be configured by one hardware component (e.g., a decoder chipset or processor) depending on the embodiment. Additionally, the memory 460 may include a decoded picture buffer (DPB) and may be configured by a digital storage medium. The hardware component may further include the memory 460 as an internal/external component.

[0083] When a bitstream containing video/image information is input, the image/video decoder 400 may reconstruct an image/video in correspondence with the process in which the image/video information is processed in the image/video encoder 300 of FIG. 3. For example, the image/video decoder 400 may derive units/blocks based on block partition-related information acquired from the bitstream. The image/video decoder 400 may perform decoding using a processing unit applied in the image/video encoder. Accordingly, the processing unit of decoding may, for example, be a coding unit, and the coding unit may be partitioned from a coding tree unit or a largest coding unit according to a quad tree structure, a binary tree structure and/or a ternary tree structure. One or more transform units may be derived from the coding unit. In addition, the reconstructed image signal decoded and output through the image/video decoder 400 may be played through a playback device.

[0084] The image/video decoder 400 may receive a signal output from the encoder of FIG. 3 in the form of a bitstream, and decode the received signal through the entropy decoder 410. For example, the entropy decoder 410 may parse the bitstream to derive information (e.g., image/video information) necessary for image reconstruction (or picture reconstruction). The image/video information may further include information about various parameter sets, such as an adaptation parameter set (APS), picture parameter set (PPS), sequence parameter set (SPS), or video parameter set (VPS). Additionally, image/video information may further include general constraint information. Additionally, the image/video information may include a method of generating and using decoded information, a purpose, and the like. The image/video decoder 400 may decode the picture further based on information about the parameter set and/or general constraint information. The signaled/received information and/or syntax elements may be decoded and acquired from the bitstream through a decoding procedure. For example, the entropy decoder 410 may decode information in the bitstream based on a coding method such as exponential Golomb coding, CAVLC, or CABAC, and output the values of syntax elements necessary for image reconstruction and quantized values of transform coefficients related to residuals. More specifically, in the CABAC entropy decoding method, a bin corresponding to each syntax element may be received in the bitstream, a context model may be determined using decoding target syntax element information and decoding information of neighboring and decoding target blocks or information on the symbol/bin decoded in the previous step, the occurrence probability of the bin may be predicted according to the determined context model, and arithmetic decoding of the bin may be performed to generate a symbol corresponding to the value of each syntax element. At this time, the CABAC entropy decoding method may update the context model using the information on the decoded symbol/bin for the context model of the next symbol/bin after determining the context model. Information about prediction among the information decoded in the entropy decoder 410 is provided to the predictor (inter predictor 432 and intra predictor 431), and a residual value obtained by performing entropy decoding in the entropy decoder 410, that is, quantized transform coefficients and related parameter information may be input to the residual processor 420. The residual processor 420 may derive a residual signal (residual block, residual samples, residual sample array). Additionally, information about filtering among the information decoded by the entropy decoder 410 may be provided to the filter 450. Meanwhile, a receiver (not shown) that receives a signal output from the image/video encoder may be further configured as an internal/external element of the image/video decoder 400, or the receiver may be a component of the entropy decoder 410. Meanwhile, the image/video decoder according to the present disclosure may be called an image/video decoding apparatus, and the image/video decoder may be divided into an information decoder (image/video information decoder) and a sample decoder (image/video sample decoder). In this case, the information decoder may include an entropy decoder 410, and the sample decoder may include at least one of a dequantizer 321, an inverse transformer 322, an adder 440, a filter 450, and a memory 460, an inter predictor 432 or an intra predictor 431.

[0085] The dequantizer 421 may dequantize the quantized transform coefficients and output transform coefficients. The dequantizer 421 may rearrange the quantized transform coefficients into a two-dimensional block form. In this case, rearranging may be performed based on the coefficient scan order performed in the image/video encoder. The dequantizer 321 may perform dequantization on quantized transform coefficients using quantization parameters (e.g., quantization step size information) and acquire transform coefficients.

[0086] The inverse transformer 422 inversely transforms the transform coefficients to acquire a residual signal (residual block, residual sample array).

[0087] The predictor 430 may perform prediction on the current block and generate a predicted block including prediction samples for the current block. The predictor may determine whether intra prediction or inter prediction is applied to the current block based on information about prediction output from the entropy decoder 410, and may determine a specific intra/inter prediction mode.

[0088] The predictor 420 may generate a prediction signal based on various prediction methods. For example, the predictor may not only apply intra prediction or inter prediction for prediction of one block, but also may apply intra prediction and inter prediction simultaneously. This may be called combined inter and intra prediction (CIIP). Additionally, the predictor may be based on an intra block copy (IBC) prediction mode or a palette mode for prediction of a block. The IBC prediction mode or palette mode may be used, for example, for image/video coding of content such as games, such as screen content coding (SCC). In IBC, prediction is basically performed within the current picture, but may be performed similarly to inter prediction in that a reference block is derived within the current picture. That is, IBC may use at least one of the inter prediction techniques described in this document. The palette mode may be viewed as an example of intra coding or intra prediction. When the palette mode is applied, information about the palette table and palette index may be included and signaled in the image/video information.

[0089] The intra predictor 431 may predict the current block by referencing samples in the current picture. The referenced samples may be located in the neighbor of the current block, or may be located away from the current block, depending on the prediction mode. In intra prediction, prediction modes may include a plurality of non-directional modes and a plurality of directional modes. The intra predictor 431 may determine the prediction mode applied to the current block using the prediction mode applied to the neighboring block.

[0090] The inter predictor 432 may derive a predicted block for the current block based on a reference block (reference sample array) specified by a motion vector in the reference picture. At this time, in order to reduce the amount of motion information transmitted in the inter prediction mode, motion information may be predicted in block, subblock, or sample units based on the correlation of motion information between the neighboring block and the current block. The motion information may include a motion vector and a reference picture index. The motion information may further include inter prediction direction (L0 prediction, L1 prediction, Bi prediction, etc.) information. In the case of inter prediction, neighboring blocks may include a spatial neighboring block present in the current picture and a temporal neighboring block present in the reference picture. For example, the inter predictor 432 may construct a motion information candidate list based on neighboring blocks and derive a motion vector and/or reference picture index of the current block based on received candidate selection information. Inter prediction may be performed based on various prediction modes, and information about prediction may include information indicating the mode of inter prediction for the current block.

[0091] The adder 440 may generate a reconstructed signal (reconstructed picture, reconstructed block, reconstructed sample array) by adding the acquired residual signal to a prediction signal (predicted block, prediction sample array) output from the predictor (including the inter predictor 432 and/or the intra predictor 431). If there is no residual for a processing target block, such as when skip mode is applied, the predicted block may be used as a reconstruction block.

[0092] The adder 440 may be called a reconstructor or a reconstruction block generator. The generated reconstructed signal may be used for intra prediction of a next processing target block in the current picture, may be output after filtering as described later, or may be used for inter prediction of a next picture.

[0093] Meanwhile, luma mapping with chroma scaling (LMCS) is applicable in a picture decoding process.

[0094] The filter 450 can improve subjective/objective image quality by applying filtering to the reconstructed signal. For example, the filter 450 may generate a modified reconstructed picture by applying various filtering methods to the reconstructed picture, and transmit the modified reconstructed picture in the memory 460, specifically the DPB of the memory 460. Various filtering methods may include, for example, deblocking filtering, sample adaptive offset, adaptive loop filter, bilateral filter, etc.

[0095] The (modified) reconstructed picture stored in the DPB of the memory 460 may be used as a reference picture in the inter predictor 432. The memory 460 may store motion information of a block from which motion information in the current picture is derived (or decoded) and/or motion information of blocks in an already reconstructed picture. The stored motion information may be transferred to the inter predictor 432 for use as motion information of spatial neighboring blocks or motion information of temporal neighboring blocks. The memory 460 may store reconstructed samples of reconstructed blocks in the current picture and transfer them to the intra predictor 431.

[0096] Meanwhile, the VCM decoder (or feature/feature map decoder) performs a series of procedures such as prediction, inverse transform, and dequantization to decode the feature/feature map, and may basically have the same/similar structure as the image/video decoder 400 described above with reference to FIG. 4. However, the VCM decoder is different from the image/video decoder 400 in that the feature/feature map is a decoding target, and may be different from the image/video decoder 400 in the name (e.g., DPB, etc.) of each unit (or component) and its specific operation. The operation of the VCM decoder may correspond to the operation of the VCM encoder, and the specific operation will be described in detail later.

Feature/feature map encoding procedure

[0097] FIG. 5 is a flowchart schematically illustrating a feature/feature map encoding procedure to which embodiments of the present disclosure are applicable.

[0098] Referring to FIG. 5, the feature/feature map encoding procedure may include a prediction procedure (S510), a residual processing procedure (S520), and an information encoding procedure (S530).

[0099] The prediction procedure (S510) may be performed by the predictor 320 described above with reference to FIG. 3.

[0100] Specifically, the intra predictor 322 may predict a current block (that is, a set of current encoding target feature elements) by referencing feature elements in a current feature/feature map. Intra prediction may be performed based on the spatial similarity of feature elements constituting the feature/feature map. For example, feature elements included in the same region of interest (RoI) within an image/video may be estimated to have similar data distribution characteristics. Accordingly, the intra predictor 322 may predict the current block by referencing the already reconstructed feature elements within the region of interest including the current block. At this time, the referenced feature elements may be located adjacent to the current block or may be located away from the current block depending on the prediction mode. Intra prediction modes for feature/feature map encoding may include a plurality of non-directional prediction modes and a plurality of directional prediction modes. The non-directional prediction modes may include, for example, prediction modes corresponding to the DC mode and planar mode of the image/video encoding procedure. Additionally, the directional modes may include prediction modes corresponding to, for example, 33 directional modes or 65 directional modes of an image/video encoding procedure. However, this is an example, and the type and number of intra prediction modes may be set/changed in various ways depending on the embodiment.

[0101] The inter predictor 321 may predict the current block based on a reference block (i.e., a set of referenced feature elements) specified by motion information on the reference feature/feature map. Inter prediction may be performed based on the temporal similarity of feature elements constituting the feature/feature map. For example, temporally consecutive features may have similar data distribution characteristics. Accordingly, the inter predictor 321 may predict the current block by referencing the already reconstructed feature elements of features temporally adjacent to the current feature. At this time, motion information for specifying the referenced feature elements may include a motion vector and a reference feature/feature map index. The motion information may further include information about an inter prediction direction (e.g., L0 prediction, L1 prediction, Bi prediction, etc.). In the case of inter prediction, neighboring blocks may include spatial neighboring blocks present within the current feature/feature map and temporal neighboring blocks present within the reference feature/feature map. A reference feature/feature map including a reference block and a reference feature/-feature map including a temporal neighboring block may be the same or different. The temporal neighboring block may be referred to as a collocated reference block, etc., and a reference feature/feature map including a temporal neighboring block may be referred to as a collocated feature/feature map. The inter predictor 321 may construct a motion information candidate list based on neighboring blocks and generate information indicating which candidate is used to derive the motion vector and/or reference feature/feature map index of the current block. Inter prediction may be performed based on various prediction modes. For example, in the case of the skip mode and the merge mode, the inter predictor 321 may use motion information of the neighboring block as motion information of the current block. In the case of the skip mode, unlike the merge mode, the residual signal may not be transmitted. In the case of the motion vector prediction (MVP) mode, the motion vector of the neighboring block is used as a motion vector predictor, and the motion vector of the current block may be indicated by signaling the motion vector difference. The predictor 320 may generate a prediction signal based on various prediction methods in addition to intra prediction and inter prediction described above.

[0102] The prediction signal generated by the predictor 320 may be used to generate a residual signal (residual block, residual feature elements) (S520). The residual processing procedure (S520) may be performed by the residual processor 330 described above with reference to FIG. 3. In addition, (quantized) transform coefficients may be generated through a transform and/or quantization procedure for the residual signal, and the entropy encoder 340 may encode information about the (quantized) transform coefficients in the bitstream as residual information (S530). Additionally, the entropy encoder 340 may encode information necessary for feature/feature map reconstruction, such as prediction information (e.g., prediction mode information, motion information, etc.), in the bitstream, in addition to the residual information.

[0103] Meanwhile, the feature/feature map encoding procedure may further include not only a procedure (S530) for encoding information for feature/feature map reconstruction (e.g., prediction information, residual information, partitioning information, etc.) and outputting it in the form of a bitstream, a procedure for generating a reconstructed feature/feature map for the current feature/feature map and a procedure (optional) for applying in-loop filtering to the reconstructed feature/feature map.

[0104] The VCM encoder may derive (modified) residual feature(s) from the quantized transform coefficient(s) through dequantization and inverse transform, and generate a reconstructed feature/feature map based on the predicted feature(s) and (modified) residual feature(s) that are the output of step S510. The reconstructed feature/feature map generated in this way may be the same as the reconstructed feature/feature map generated in the VCM decoder. When an in-loop filtering procedure is performed on the reconstructed feature/feature map, a modified reconstructed feature/-feature map may be generated through the in-loop filtering procedure on the reconstructed feature/feature map. The modified reconstructed feature/feature map may be stored in a decoded feature buffer (DFB) or memory and used as a reference feature/feature map in the feature/feature map prediction procedure later. Additionally, (in-loop) filtering-related information (parameters) may be encoded and output in the form of a bitstream. Through the in-loop filtering procedure, noise that may occur during feature/feature map coding may be removed, and feature/feature map-based task performance may be improved. In addition, by performing an in-loop filtering procedure at both the encoder stage and the decoder stage, the identity of the prediction result can be guaranteed, the reliability of feature/feature map coding can be improved, and the amount of data transmission for feature/feature map coding can be reduced.

Feature/feature map decoding procedure

[0105] FIG. 6 is a flowchart schematically illustrating a feature/feature map decoding procedure to which embodiments of the present disclosure are applicable.

[0106] Referring to FIG. 6, the feature/feature map decoding procedure may include an image/video information acquisition procedure (S610), a feature/feature map reconstruction procedure (S620 to S640), and an in-loop filtering procedure for a reconstructed feature/feature map (S650). The feature/feature map reconstruction procedure may be performed on the prediction signal and residual signal acquired through inter/intra prediction (S620) and residual processing (S630), dequantization and inverse transform process for quantized transform coefficients described in the present disclosure. A modified reconstructed feature/feature map may be generated through an in-loop filtering procedure for the reconstructed feature/feature map, and the modified reconstructed feature/feature map may be output as a decoded feature/feature map. The decoded feature/feature map may be stored in a decoded feature buffer (DFB) or memory and used as a reference feature/feature map in the inter prediction procedure when decoding the feature/feature map. In some cases, the above-described in-loop filtering procedure may be omitted. In this case, the reconstructed feature/feature map may be output without change as a decoded feature/feature map, and stored in the decoded feature buffer (DFB) or memory, and then be used as a reference feature/feature map in the inter prediction procedure when decoding the feature/feature map.

Embodiments

[0107] A conventional feature encoding/decoding apparatus is affected by an output result of a feature extraction network. Therefore, it is impossible to independently develop and apply a feature encoding/decoding method.

[0108] FIG. 7 is a view showing an example of a conventional feature encoding and decoding structure. Referring to FIG. 7, an input of a feature encoding apparatus (VCM encoder) may be feature information (feature/video/image) extracted from a feature extraction network. When the feature extraction network is changed or an extraction method or an extraction position is changed, a property of feature information, which is input into the feature encoding apparatus, may also be changed. However, because an operation of the feature encoding apparatus and a feature decoding apparatus (VCM decoder) is defined in advance, changed feature information (e.g., a data type, a range, etc.) may go beyond a permissible input range of the feature encoding apparatus and the feature decoding apparatus, and in this case, the feature encoding apparatus and the feature decoding apparatus may not operate normally. Accordingly, a method for responding variability of a feature extraction network is needed.

[0109] Meanwhile, a method for responding variability of a feature extraction network is also needed to satisfy the following requirements (Table 1) that VCM shall operate independently of a network model and a machine task.

[Table 1]

VCM shall operate independently of the network models. (Video/Feature coding) VCM shall operate independently of the machine task types. (Video/Feature coding) Description: VCM shall support multiple type of machine tasks. (Feature/Video coding)

[0110] This application proposes various methods for responding variability of a feature extraction network through embodiments described below.

Embodiment 1

[0111] Embodiment 1 is an example for responding variability of a feature extraction network by using transform (input transform) of feature information.

[0112] For the VCM encoder 222 and the VCM decoder 223 to operate independently of a machine task and the feature extraction network 221, an input of the VCM encoder 222 should be irrelevant to a change in the feature extraction network 221. That is, feature information input into the VCM encoder 222 should have a predefined range and form, and herein the predefined range and form may be within a permissible range of the VCM encoder 222 and/or the VCM decoder 223.

[0113] However, as described above, the feature extraction network 221 may be changed or an extraction position of a feature may be changed, and thus the range and form of extracted feature information may be changed. Accordingly, even when the range and form of extracted feature information are changed, in order to input the feature information into the VCM encoder 222 and/or the VCM decoder 223 in a predefined range and a predefined form, a process of transforming the extracted feature information into the predefined range and the predefined form is needed.

[0114] FIG. 8 is a view showing an example of a feature encoding and decoding structure to which embodiments of the present disclosure are applicable.

[0115] Referring to FIG. 8, feature transform 224 may perform a function of transforming feature information (extracted feature information) output from the feature extraction network 221 into a predefined range and a predefined form. An input of the feature transform 224 may be feature information extracted from the feature extraction network 221, and an output may be transformed feature information (data) and information (transform information) necessary for inverse transform. Here, the information necessary for inverse transform may be 'information on transform'.

[0116] As an example, feature transform may be a process of quantizing extracted feature information. In the example, a data type of feature information extracted through the feature extraction network 221 may have various ranges and forms such as float, integer, signed and unsigned. If the VCM encoder 222 permits only unsigned char and unsigned integer for inputs, extracted feature information may be quantized in a range (or type) permitted by the VCM encoder 222.

[0117] In this case, transformed feature information may be a quantized feature map, and information on transform (information necessary for inverse transform) may be information necessary for dequantization. The information necessary for dequantization may include a quantization method used for transform and supplementary information (e.g., in the case of uniform quantization, Min & Max values and bit).

[0118] An encoded bitstream may consist of a feature coding layer payload, which is an encoded result of transformed feature information (data), and transform information that is information on transform. The VCM decoder 223 may independently decode the feature coding layer payload, and the decoded result (decoded feature data, obtained feature information) and the information on transform may be inputs of feature inverse transform 225. The feature inverse transform 225 may reconstruct (inversely transform) feature information in the range and form of original feature information (feature information output from a feature extraction network) based on the information on feature. The reconstructed feature information may be used as an input of a machine task.

[0119] FIG. 9 is a flowchart showing a feature encoding method according to Embodiment 1.

[0120] Referring to FIG. 9, the feature encoding apparatus 10 may obtain feature information of input data from the input data (S910). Herein, the obtaining of the feature information may be extracting the feature information from the input data.

[0121] The feature encoding apparatus 10 may transform the feature information and generate the transformed feature information (S920). The feature information may be transformed in a predefined range and a predefined form, and the predefined range and form may be within a permissible range of the VCM encoder 222 and/or the VCM decoder 223. The transformed feature information may be generated by quantizing the feature information.

[0122] The feature encoding apparatus 10 may encode the transformed feature information and information on transform (S930). The information on transform may be information necessary to inversely transform the transformed feature information. The information on transform may include multiple pieces of information related to the transforming of the feature information. For example, the information on transform may include information on a transform type that is used to transform the feature information.

[0123] FIG. 10 is a flowchart showing a feature decoding method according to Embodiment 1.

[0124] Referring to FIG. 10, the feature decoding apparatus 20 may obtain information on transform and feature information from a bitstream (S1010). The feature information may be 'the transformed feature information' of the process S920. The information on transform may be information necessary to inversely transform the transformed feature information. The information on transform may include multiple pieces of information related to the transforming of the feature information. For example, the information on transform may include information on a transform type, and the information on the transform type may represent a transform type that is used to transform the feature information.

[0125] The feature decoding apparatus 20 may inversely transform the feature information based on the information on transform (S1020). The inversely transformed feature information may be the extracted feature information of the process S910. That is, the inverse transform process of S1020 may be a process of inversely transforming transformed feature information into original feature information in response to the transform process of S910. The inversely transformed feature information may be generated by dequantizing the feature information. The inversely transformed feature information may be used as an input of a machine task.

Embodiment 2

[0126] Embodiment 2 is an embodiment related to a method of determining a type for transforming and inversely transforming feature information.

[0127] FIG. 11 is a flowchart showing a feature encoding method according to Embodiment 2.

[0128] Referring to FIG. 11, the feature encoding apparatus 10 may determine a type for transforming extracted feature information, that is, a transform type (S1110). For example, the feature encoding apparatus 10 may determine any one of predetermined candidate transform types as a transform type of extracted feature information. The candidate transform types may be types capable of transforming feature information within a permissible range of the feature encoding apparatus 10 and the feature decoding apparatus 20 even when the feature information is changed. Among the candidate transform types, a transform type capable of transforming the range, type and property of feature information within a permissible range may be determined as a transform type of the feature information. Information on the determined transform type may be encoded in a bitstream.

[0129] The feature encoding apparatus 10 may transform the feature information based on the determined transform type (S1120). For example, the feature encoding apparatus 10 may quantize the feature information based on a determined quantization type.

[0130] FIG. 12 is a flowchart showing a feature decoding method according to Embodiment 2.

[0131] Referring to FIG. 12, the feature decoding apparatus 20 may obtain information on a transform type from a bitstream. In addition, the feature decoding apparatus 20 may determine or identify a transform type applied to feature information based on the information on the transform type (S1210). The process S1210 may be a process of identifying an inverse transform type to be applied to the feature information among predetermined candidate (inverse) transform types based on the information on the transform type.

[0132] The feature decoding apparatus 20 may inversely transform the feature information based on the determined or identified inverse transform type (S1220). Through the process S1220, feature information corresponding to feature information extracted from input data may be derived.

[0133] FIG. 13 is a view for explaining an example of determining a transform type and an inverse transform type based on Embodiment 2.

[0134] A data classifier of feature transform 224 may perform a function of determining a transformer suitable for input feature information (extracted feature information). Transformer A to Transformer D may correspond to candidate transform types, and the determined transformer may correspond to a determined transform type.

[0135] When the suitable transformer is determined, information on the transform type for identifying the determined transformer and information for inverse transform may be stored. In addition, the information on the transform type and the information for inverse transform may be encoded in a bitstream. The information on the transform type may be implemented by data_property_id described below.

[0136] Feature inverse transform 225 may identify an inverse transformer to be applied to feature information among predetermined candidate inverse transform types by using data_property_id. Herein, Inverse Transformer A to Inverse Transformer D may correspond to candidate inverse transform types, and the identified inverse transformer may correspond to an identifier inverse transform type. Feature inverse transform may inversely transform feature information into the range and form of original feature information by using the feature information and information necessary for inverse transform.

[0137] Table 2 shows an example of data_property_id that represents a property of input feature information.

[Table 2]

data_property_id	Data Property	Transformer
00	float	Quantization A
01	unsigned float	Quantization B
10	integer	Quantization C
11	unsigned integer	Quantization D

[0138] In Table 2, Data Property represents a property of data, and apart from a type of data, a resolution or a distribution characteristic may also be expressed as a property of data. Table 2 defines 4 data properties, and 4 transformers are also defined correspondingly. Information necessary for inverse transform may be defined for each of the defined transformers.

Embodiment 3

[0139] Embodiment 3 is an embodiment of a syntax structure for supporting a proposed method of this application for transforming and inversely transforming feature information.

[0140] Information on transform may consist of a feature transform header feature_transform_header and a feature transform rbsp feature_transform_rbsp. feature_transform_header may include whether or not transform is applied and information on a range thereof, and feature_transform_rbsp may include information necessary for transform (inverse transform) defined in feature_transform_header.

[0141] Table 3 shows an example of a syntax structure of feature_transform_header.

[Table 3]

Feature_transform_header() {	**Descriptor**
transform_bypass_flag	u(1)
if(transform_bypass_flag){	
transform_level_id	u(2)
}	
}	

[0142] transform_bypass_flag may indicate whether or not feature transform is applied. transform_bypass_flag with a value of 0 may indicate that feature information is transformed, and transform_bypass_flag with a value of 1 may indicate that feature information is not transformed. When the value of transform_bypass_flag is 1, an output of the feature extraction network 221 may be transferred without change to the VCM encoder 222. transform_bypass_flag may be referred to as 'first information'.

[0143] In case feature information is transformed to encode original feature information, information for identifying a transform unit may be needed. transform_level_id may indicate an application unit of transform. transform_level_id with a value of 0 may indicate that a same transform type and information on transform are applied at a sequence level, and transform_level_id with a value of 1 may indicate that a same transform type and information on transform are applied at a feature map level (frame level). In addition, transform_level_id with a value of 2 may indicate that a same transform type and information on transform are applied at a channel level. transform_level_id may be referred to as 'second information'.

[0144] A sequence unit, a frame unit and a channel unit have been described as examples where transform is applied, but transform and inverse transform of feature information may also be applied in another unit, and in such a case, information for identifying the unit of transform and inverse transform may be required.

[0145] Table 4 shows an example of a syntax structure of feature_transform_rbsp.

[Table 4]

Feature_transform_rbsp() {	**Descriptor**
data_property_id	u(2)
transform_bitdepth	u(v)
transform_width	u(v)
transform_height	u(v)
If(transform_level_id>0)	
Frame_unique_id	u(v)
if(data_property_id==0){	
property_id_0_information()	
}else if(data_property_id==1){	
property_id_1_information()	
}else if(data_property_id==2){	
property_id_2_information()	
}else if(data_property_id==3){	
property_id_3_information()	
}	
rbsp_trailing_bits()	
}	

[0146] data_property_id may indicate a property of original feature and be information on a transform type. Embodiment 2 describes a case where data_property_id indicates a total of 4 values, but a range of values, which data_property_id may indicate, may vary depending on expansion of property. A property of feature information and a transform type according to the property may be defined in advance. A transform type and an inverse transform type applied to feature information may be determined according to a value of data_property_id.

[0147] transform_bitdepth may indicate a bit depth of transformed feature information (transformed feature map). The bit depth of transformed feature information may have a value within a predefined dynamic range of the VCM encoder 222. transform_width and transform_height may indicate a horizontal size and a vertical size of transformed feature information (transformed feature map). transform_bitdepth, transform_width and transform_height may be utilized for the feature encoding apparatus 10 to obtain information on an encoding target.

[0148] Information for identifying a frame or feature in a feature coding layer, to which information on transform corresponds, may be needed to enable the operation of the VCM encoder 222 and the VCM decoder 223 to be performed independently of feature transform. Frame_unique_id may indicate a feature map to which transform is applied or a feature channel to which transform is applied. That is, Frame_unique_id may correspond to information for identifying a frame or feature in a feature coding layer to which information on transform corresponds. Frame_unique_id may be referred to as 'third information'.

[0149] property_id_0_information, property_id_1_information, property_id_2_information and property_id_3_information may be supplementary information corresponding to each transform type. The supplementary information may correspond to information necessary for inverse transform of feature information. In this application, 4 pieces of supplementary information are defined as a result of definition of 4 properties for feature information, but the number of supplementary information may be expanded along with expansion of the feature information.

[0150] Table 5 shows an example of a syntax structure of supplementary information (property_id_0_information).

[Table 5]

Property_id_0_information( ) {	**Descriptor**
total_channel_number	u(v)
Input_bitdepth	u(v)
if(trasfrom_level_id ==0){	
min_value	u(v)
max_value	u(v)
}else **if(trasfrom_level_id ==1){**	
min_value	u(v)
max_value	u(v)
}else if(trasfrom_level_id ==2){	
for(i = 0; i < total_channel_number; i++){	
min_value[i]	u(v)
max_value[i]	u(v)
}	
}	
}	

[0151] Input_bitdepth may indicate a bit depth of original feature information, and min_value and max_value may indicate a minimum value and a maximum value of the original feature information, respectively. In this embodiment, as quantization is described as an example of transform, supplementary information necessary for inverse transform is described to include a bit depth, a minimum value and a maximum value. However, in case a different transform method from quantization is applied, parameters for inverse transform of the method may be included in supplementary information.

[0152] FIG. 14 is a flowchart showing a feature encoding method according to Embodiment 3.

[0153] The feature encoding apparatus 10 may determine whether or not to apply feature transform (S1410). When determining not to apply feature transform, the feature encoding apparatus 10 may encode the value of transform_bypass_flag as 1 (S1420).

[0154] On the other hand, when determining to apply feature transform, the feature encoding apparatus 10 may determine a unit (or level) where feature transform is applied (S1430). For example, the feature encoding apparatus 10 may determine whether or not to apply feature transform to a level equal to or below a feature map level.

[0155] The feature encoding apparatus 10 may encode information on transform in a bitstream (S1450 and S1440). For example, if it is determined that feature transform is not applied to a level equal to or below the feature map level, the value of transform_bypass_flag may be encoded as 1, and transform_level_id and data_property_id may be encoded. As another example, if it is determined that feature transform is applied to the level equal to or below the feature map level, the value of transform_bypass_flag may be encoded as 1, transform_level_id and data_property_id may be encoded, and Frame_unique_id may also be encoded. That is, Frame_unique_id may be encoded only when it is determined that feature transform is applied to the level equal to or below the feature map level.

[0156] In addition, the feature encoding apparatus 10 may select supplementary information corresponding to a value of data_property_id and encode the selected supplementary information. As described above, the supplementary information may include a variety of information for inverse transform of feature information.

[0157] FIG. 15 is a flowchart showing a feature decoding method according to Embodiment 3.

[0158] Referring to FIG. 15, the feature decoding apparatus 20 may obtain transform_bypass_flag from a bitstream and determine a value of transform_bypass_flag (S1510). The process S1510 may be a process of determining, based on the value of transform_bypass_flag, whether or not to apply inverse transform (that is, whether or not transform is applied).

[0159] If the value of transform_bypass_flag is 1, transform is not applied to feature information, and inverse transform may not be applied to the feature information. On the other hand, if the value of transform_bypass_flag is 0, transform is applied to the feature information, and inverse transform may be applied to the feature information.

[0160] If the value of transform_bypass_flag is 0, the feature decoding apparatus 20 may obtain transform_level_id and data_property_id from a bitstream (S1520). In addition, the feature decoding apparatus 20 may determine whether or not the value of transform_level_id exceeds 0 (S1530) and obtain Frame_unique_id from a bitstream, when the value of transform_level_id exceeds 0 (S1540).

[0161] In addition, the feature decoding apparatus 20 may obtain supplementary information corresponding to the value of data_property_id from a bitstream based on the value of data_property_id and inversely transform the feature information by using the supplementary information.

[0162] FIG. 16 is a flowchart showing an example of each of feature information and information on transform at each step of feature encoding and decoding. Referring to 16, input and output information of each step may be described as follows.

1) Feature extraction network step

[0163] This is a step of extracting feature information (feature map) through the feature extraction network 221, and the property of the feature map may be {Data type: float, Bit depth: 32, Width: 1920, Height: 1080, Channel: 64}.

2) Feature transform step

[0164] This is a step of transforming the feature map, which is an output of the feature extraction network 221, to be suitable for an operating range of the VCM encoder 222, and information on transform according to the property of the feature map may be constructed. First, whether or not transform is needed is determined and the value of transform_bypass_flag of feature_transform_header is defined, and if application of transform is needed, the value of transform_level_id indicating a range of transform may be defined.

[0165] When transform is defined at a frame level, feature_transform_header may be defined as {transform_bypass_flag: 0, transform_level_id: 1}. In Feature_transform_rbsp, supplementary information necessary for inverse transform may be defined as {total_channel_number: 64, Input_bitdepth: 32, min_value: -0.1, max_value: 12}, and property information of transformed feature information may be defined. According to the information on transform, a transform type is selected so that the feature map is transformed to be suitable for an input range of the VCM encoder 222.

3) VCM encoding step

[0166] Encoding may be performed using the transformed feature map and the information on transform.

4) VCM decoding step

[0167] The feature map and the information on transform may be decoded. The decoded feature map may correspond to the transformed feature map.

5) Feature inverse transform step

[0168] Inverse transform for the decoded feature map (transformed feature map) may be performed based on the information on transform. Finally, a reconstructed feature map with a same property as an output of the feature extraction network 221 may be output.

[0169] While the exemplary methods of the present disclosure described above are represented as a series of operations for clarity of description, it is not intended to limit the order in which the steps are performed, and the steps may be performed simultaneously or in different order as necessary. In order to implement the method according to the present disclosure, the described steps may further include other steps, may include remaining steps except for some of the steps, or may include other additional steps except for some steps.

[0170] In the present disclosure, the image encoding apparatus or the image decoding apparatus that performs a predetermined operation (step) may perform an operation (step) of confirming an execution condition or situation of the corresponding operation (step). For example, if it is described that predetermined operation is performed when a predetermined condition is satisfied, the image encoding apparatus or the image decoding apparatus may perform the predetermined operation after determining whether the predetermined condition is satisfied.

[0171] The various embodiments of the present disclosure are not a list of all possible combinations and are intended to describe representative aspects of the present disclosure, and the matters described in the various embodiments may be applied independently or in combination of two or more.

[0172] Embodiments described in the present disclosure may be implemented and performed on a processor, microprocessor, controller, or chip. For example, the functional units shown in each drawing may be implemented and performed on a computer, processor, microprocessor, controller, or chip. In this case, information for implementation (e.g., information on instructions) or algorithm may be stored in a digital storage medium.

[0173] In addition, the decoder (decoding apparatus) and the encoder (encoding apparatus), to which the embodiment(s) of the present disclosure are applied, may be included in a multimedia broadcasting transmission and reception device, a mobile communication terminal, a home cinema video device, a digital cinema video device, a surveillance camera, a video chat device, a real time communication device such as video communication, a mobile streaming device, a storage medium, a camcorder, a video on demand (VoD) service providing device, an OTT video (over the top video) device, an Internet streaming service providing device, a three-dimensional (3D) video device, an argument reality (AR) device, a video telephony video device, a transportation terminal (e.g., vehicle (including autonomous vehicle) terminal, robot terminal, airplane terminal, ship terminal, etc.) and a medical video device, and the like, and may be used to process video signals or data signals. For example, the OTT video devices may include a game console, a blu-ray player, an Internet access TV, a home theater system, a smartphone, a tablet PC, a digital video recorder (DVR), or the like.

[0174] Additionally, a processing method to which the embodiment(s) of the present disclosure is applied may be produced in the form of a program executed by a computer and stored in a computer-readable recording medium. Multimedia data having a data structure according to the embodiment(s) of this document may also be stored in a computer-readable recording medium. Computer-readable recording media include all types of storage devices and distributed storage devices that store computer-readable data. Computer-readable recording media include, for example, Blu-ray Disc (BD), Universal Serial Bus (USB), ROM, PROM, EPROM, EEPROM, RAM, CD-ROM, magnetic tape, floppy disk, and optical data storage device. Additionally, computer-readable recording media include media implemented in the form of carrier waves (e.g., transmission via the Internet). Additionally, the bitstream generated by the encoding method may be stored in a computer-readable recording medium or transmitted through a wired or wireless communication network.

[0175] Additionally, the embodiment(s) of the present disclosure may be implemented as a computer program product by program code, and the program code may be executed on a computer by the embodiment(s) of the present disclosure. The program code may be stored on a carrier readable by a computer.

[0176] FIG. 20 is a view illustrating an example of a content streaming system to which embodiments of the present disclosure are applicable.

[0177] Referring to FIG. 20, the content streaming system, to which the embodiment of the present disclosure is applied, may largely include an encoding server, a streaming server, a web server, a media storage, a user device, and a multimedia input device.

[0178] The encoding server compresses contents input from multimedia input devices such as a smartphone, a camera, a camcorder, etc. into digital data to generate a bitstream and transmits the bitstream to the streaming server. As another example, when the multimedia input devices such as smartphones, cameras, camcorders, etc. directly generate a bitstream, the encoding server may be omitted.

[0179] The bitstream may be generated by an image encoding method or an image encoding apparatus, to which the embodiment of the present disclosure is applied, and the streaming server may temporarily store the bitstream in the process of transmitting or receiving the bitstream.

[0180] The streaming server transmits the multimedia data to the user device based on a user's request through the web server, and the web server serves as a medium for informing the user of a service. When the user requests a desired service from the web server, the web server may deliver it to a streaming server, and the streaming server may transmit multimedia data to the user. In this case, the contents streaming system may include a separate control server. In this case, the control server serves to control a command/response between devices in the contents streaming system.

[0181] The streaming server may receive contents from a media storage and/or an encoding server. For example, when the contents are received from the encoding server, the contents may be received in real time. In this case, in order to provide a smooth streaming service, the streaming server may store the bitstream for a predetermined time.

[0182] Examples of the user device may include a mobile phone, a smartphone, a laptop computer, a digital broadcasting terminal, a personal digital assistant (PDA), a portable multimedia player (PMP), navigation, a slate PC, tablet PCs, ultrabooks, wearable devices (e.g., smartwatches, smart glasses, head mounted displays), digital TVs, desktops computer, digital signage, and the like.

[0183] Each server in the contents streaming system may be operated as a distributed server, in which case data received from each server may be distributed.

[0184] FIG. 21 is a diagram illustrating another example of a content streaming system to which embodiments of the present disclosure are applicable.

[0185] Referring to FIG. 21, in an embodiment such as VCM, a task may be performed in a user terminal or a task may be performed in an external device (e.g., streaming server, analysis server, etc.) according to the performance of the device, the user's request, the characteristics of the task to be performed, etc. In this way, in order to transmit information necessary to perform a task to an external device, the user terminal may generate a bitstream including information necessary to perform the task (e.g., information such as task, neural network and/or usage) directly or through an encoding server.

[0186] In an embodiment, the analysis server may perform a task requested by the user terminal after decoding the encoded information received from the user terminal (or from the encoding server). At this time, the analysis server may transmit the result obtained through the task performance back to the user terminal or may transmit it to another linked service server (e.g., web server). For example, the analysis server may transmit a result obtained by performing a task of determining a fire to a fire-related server. In this case, the analysis server may include a separate control server. In this case, the control server may serve to control a command/response between each device associated with the analysis server and the server. In addition, the analysis server may request desired information from a web server based on a task to be performed by the user device and the task information that may be performed. When the analysis server requests a desired service from the web server, the web server transmits it to the analysis server, and the analysis server may transmit data to the user terminal. In this case, the control server of the content streaming system may serve to control a command/response between devices in the streaming system.

Industrial Applicability

[0187] The embodiments of the present disclosure may be used to encode or decode a feature/feature map.

Claims

1. A feature decoding method performed by a feature decoding apparatus, the feature decoding method comprising:

    obtaining information on transform of feature information and the feature information from a bitstream; and
    inversely transforming the feature information based on the information on the transform,
    wherein the information on the transform comprises information on a transform type that is used to transform the feature information.

2. The feature decoding method of claim 1, wherein the inversely transforming comprises:

    identifying an inverse transform type indicated by the information on the transform type among predetermined candidate transform types; and
    inversely transforming the feature information based on the identified inverse transform type.

3. The feature decoding method of claim 2, wherein the information on the transform further comprises supplementary information corresponding to the identified transform type, and
    wherein the feature information is inversely transformed based on the identified transform type and the supplementary information.

4. The feature decoding method of claim 1, wherein the transform applied to the feature information is quantization of the feature information based on the transform type, and
    wherein the inversely transforming dequantizes the feature information.

5. The feature decoding method of claim 1, wherein the information on the transform comprises first information indicating whether or not transform is applied to the feature information, and
    wherein the information on the transform type is included in the information on the transform based on the first information indicating that the transform is applied to the feature information.

6. The feature decoding method of claim 5, wherein based on the first information indicating that the transform is applied to the feature information, the information on the transform comprises second information indicating an application unit of the transform, and
    wherein the application unit of the transform comprises at least one of a sequence level, a feature map level or a channel level.

7. The feature decoding method of claim 6, wherein based on the second information indicating a level equal to or below the feature map level, the information on the transform comprises third information indicating a feature map to which the transform is applied or a feature channel to which the transform is applied.

8. A feature encoding method performed by a feature encoding apparatus, the feature encoding method comprising:

obtaining feature information of input data from the input data;
generating transformed feature information by transforming the feature information; and
encoding the transformed feature information and information on transform,
wherein the information on the transform comprises information on a transform type that is used to transform the feature information.

9. A computer-readable recording medium storing a bitstream generated by the feature encoding method of claim 8.

10. A method for transmitting a bitstream generated by a feature encoding method, wherein the feature encoding method comprises:

obtaining feature information of input data from the input data;
generating transformed feature information by transforming the feature information; and
encoding the transformed feature information and information on transform, and
wherein the information on the transform comprises information on a transform type that is used to transform the feature information.

FIG. 1

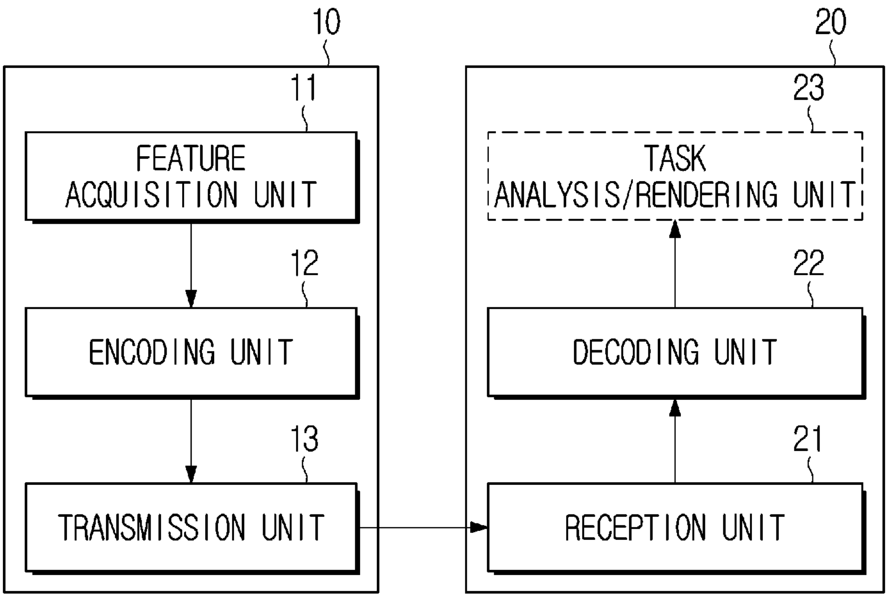

10
11
FEATURE ACQUISITION UNIT
12
ENCODING UNIT
13
TRANSMISSION UNIT
20
23
TASK ANALYSIS/RENDERING UNIT
22
DECODING UNIT
21
RECEPTION UNIT

FIG. 2

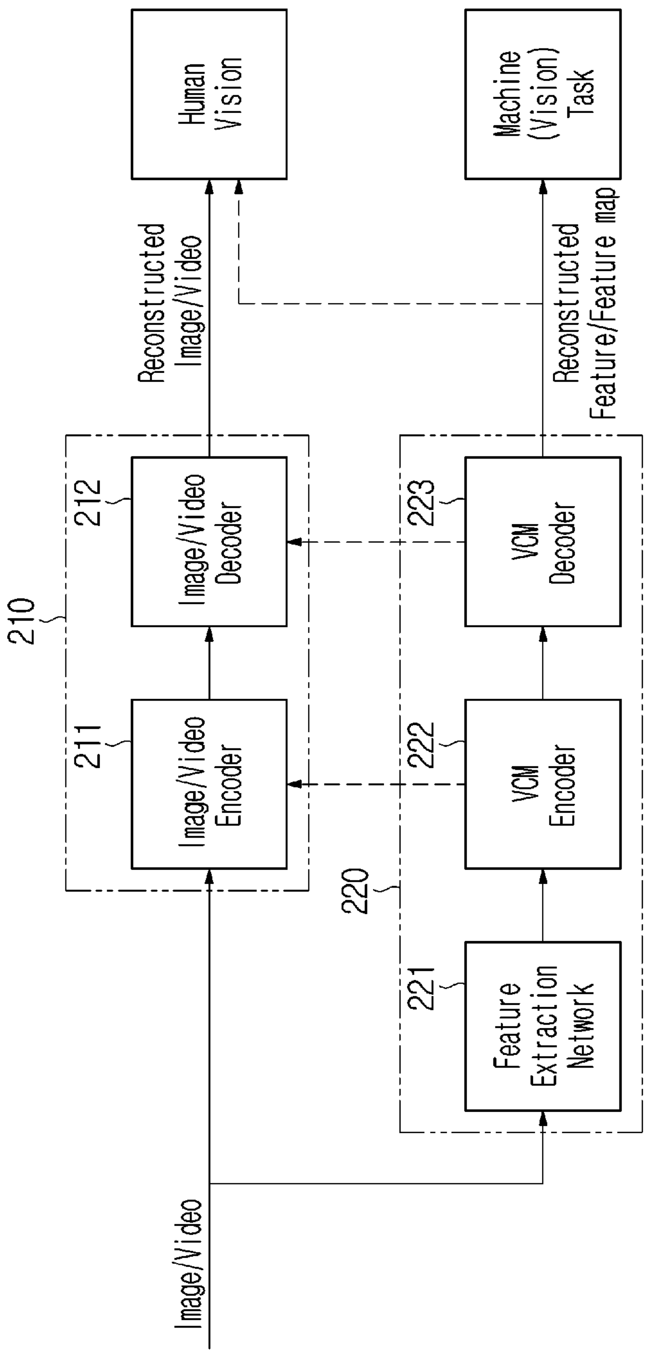

Image/Video
210
211
Image/Video Encoder
212
Image/Video Decoder
Reconstructed Image/Video
Human Vision
220
221
Feature Extraction Network
222
VCM Encoder
223
VCM Decoder
Reconstructed Feature/Feature map
Machine (Vision) Task

FIG. 3

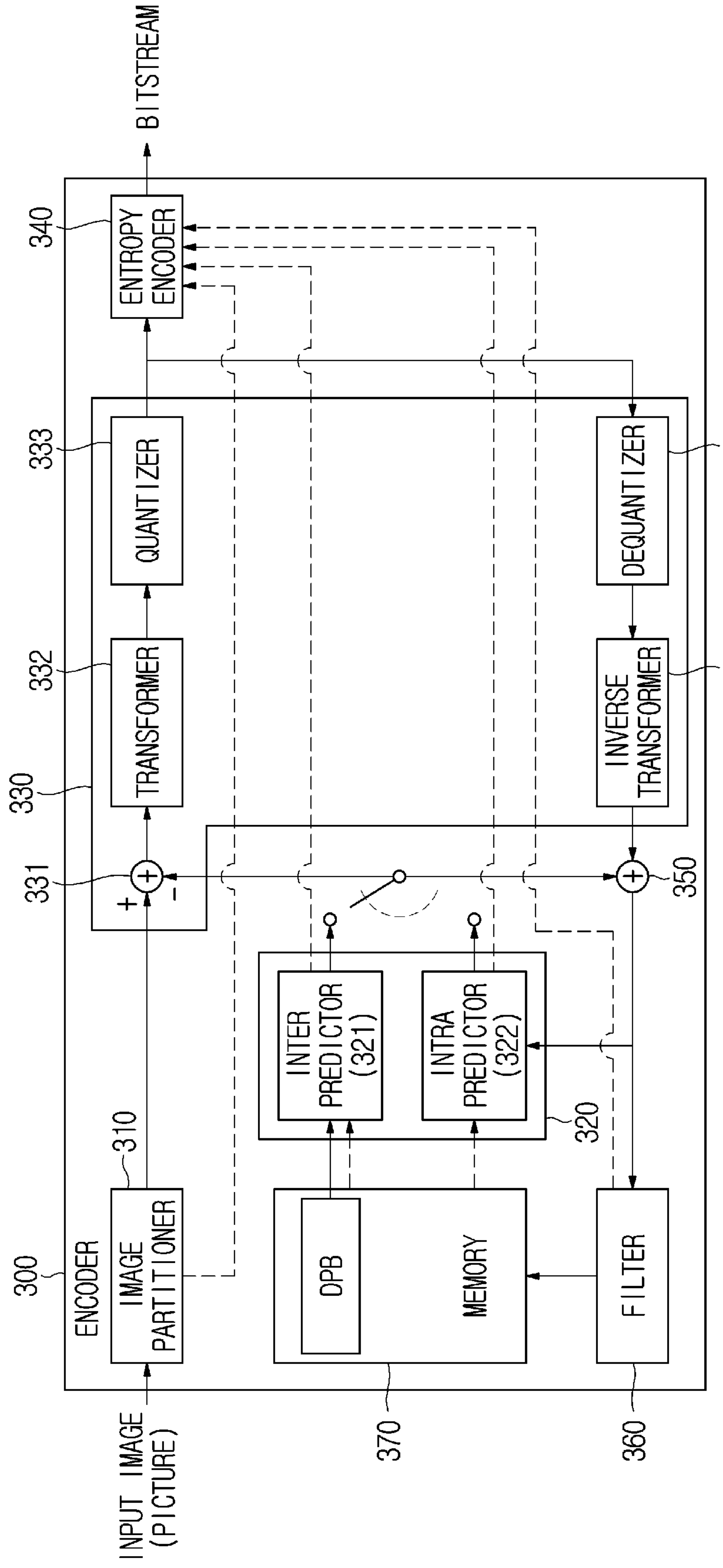

BITSTREAM
300
ENCODER
310
IMAGE PARTITIONER
320
INTER PREDICTOR (321)
INTRA PREDICTOR (322)
330
331
332
TRANSFORMER
333
QUANTIZER
334
DEQUANTIZER
335
INVERSE TRANSFORMER
340
ENTROPY ENCODER
350
360
FILTER
370
DPB
MEMORY
INPUT IMAGE (PICTURE)

FIG. 4

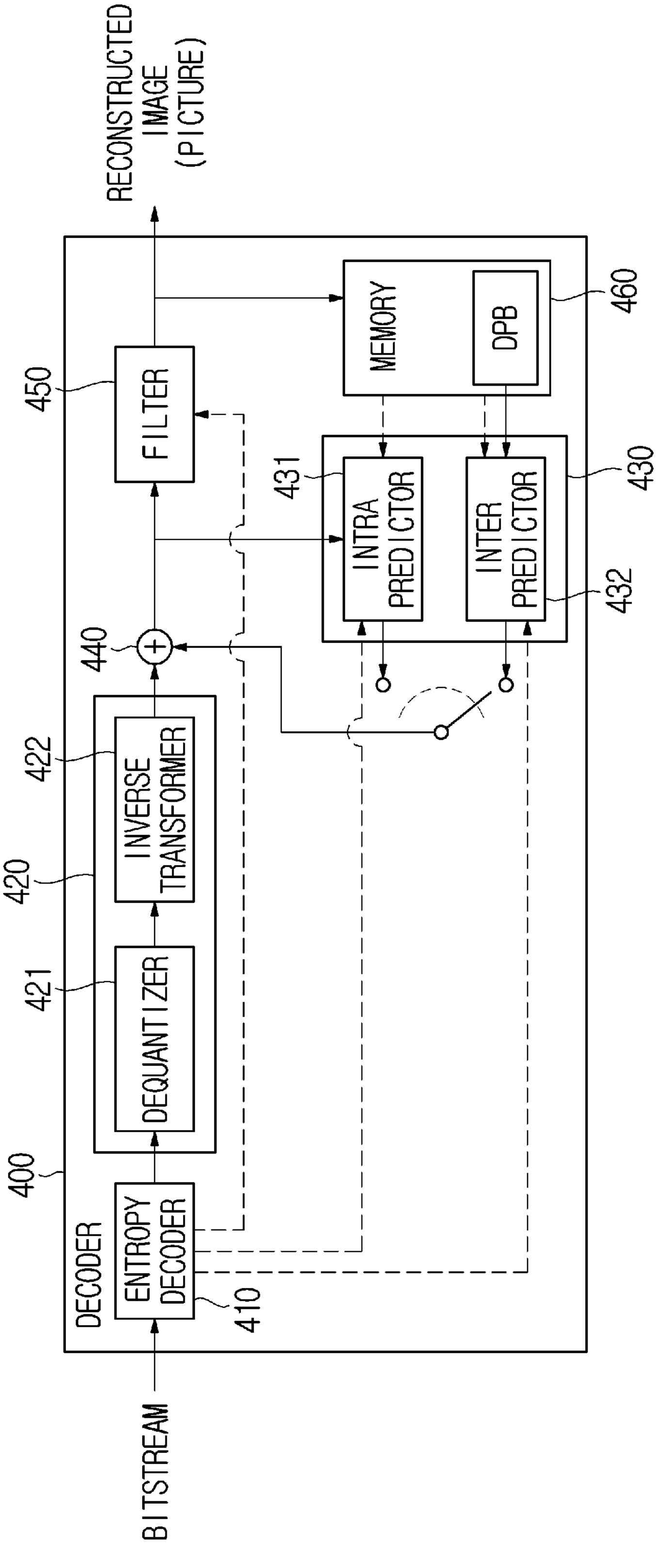

RECONSTRUCTED IMAGE (PICTURE)
BITSTREAM
DECODER
400
ENTROPY DECODER
410
420
DEQUANTIZER
421
INVERSE TRANSFORMER
422
440
FILTER
450
MEMORY
DPB
460
430
INTRA PREDICTOR
431
INTER PREDICTOR
432

FIG. 5

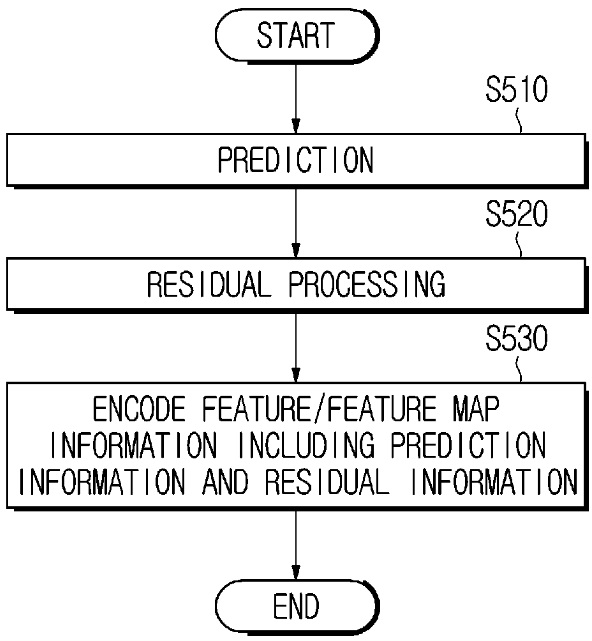

START
S510
PREDICTION
S520
RESIDUAL PROCESSING
S530
ENCODE FEATURE/FEATURE MAP INFORMATION INCLUDING PREDICTION INFORMATION AND RESIDUAL INFORMATION
END

FIG. 6

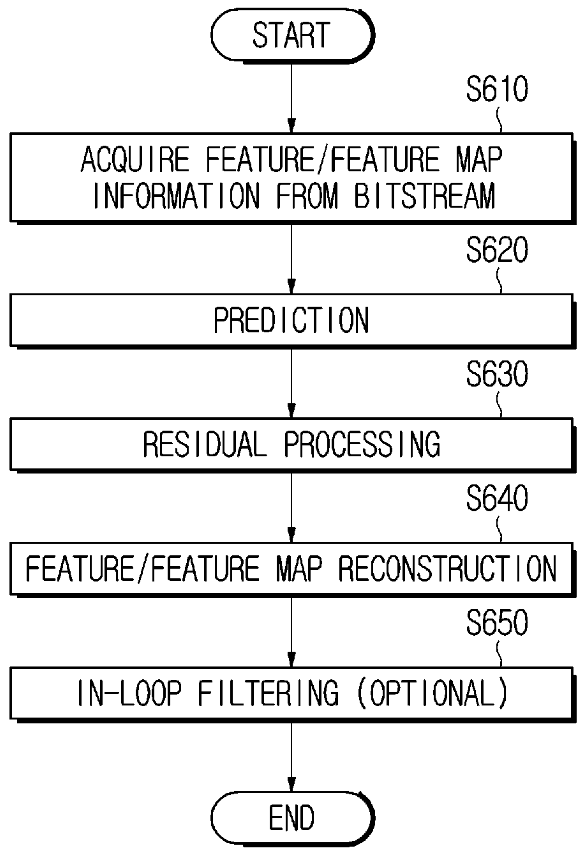

START
S610
ACQUIRE FEATURE/FEATURE MAP INFORMATION FROM BITSTREAM
S620
PREDICTION
S630
RESIDUAL PROCESSING
S640
FEATURE/FEATURE MAP RECONSTRUCTION
S650
IN-LOOP FILTERING (OPTIONAL)
END

FIG. 7

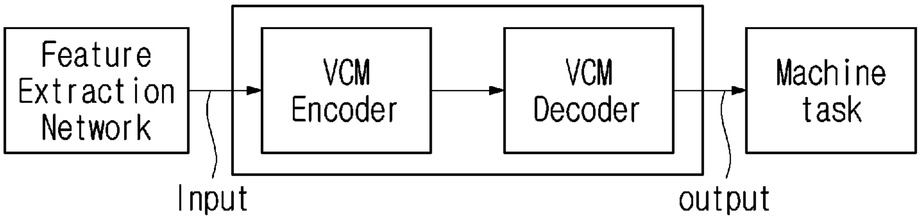

Feature Extraction Network
VCM Encoder
VCM Decoder
Machine task
Input
output

FIG. 8

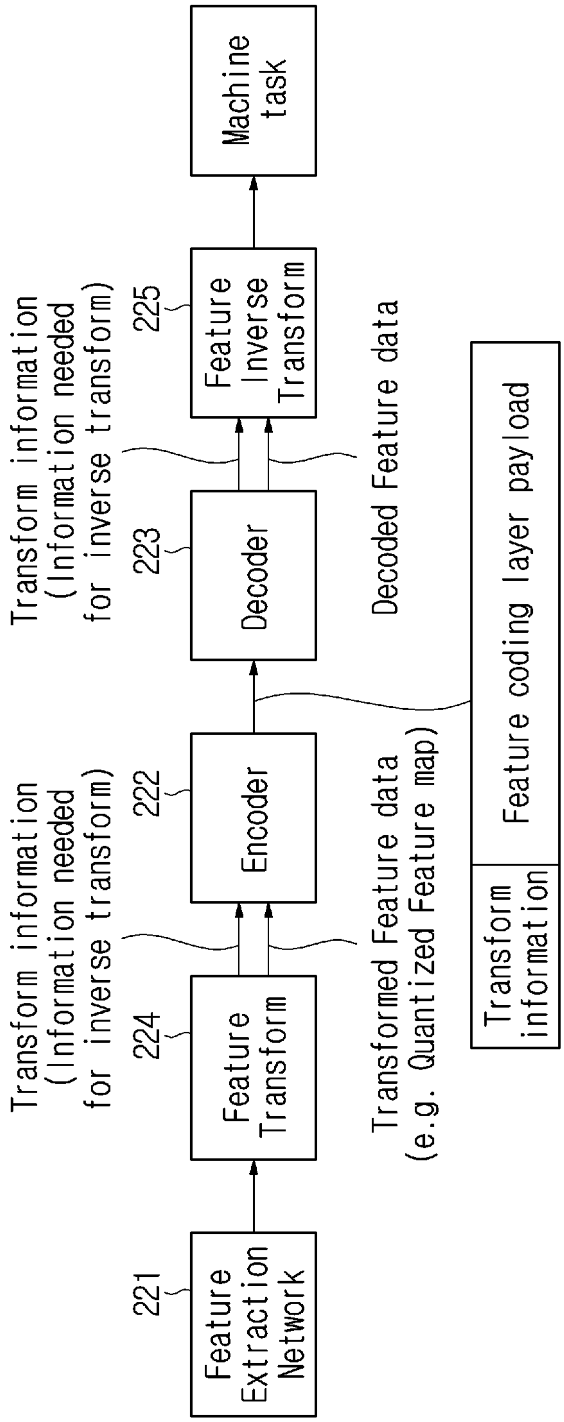

221
Feature Extraction Network
224
Feature Transform
Transform information (Information needed for inverse transform)
Transformed Feature data (e.g. Quantized Feature map)
222
Encoder
223
Decoder
Transform information (Information needed for inverse transform)
Decoded Feature data
225
Feature Inverse Transform
Machine task
Transform information
Feature coding layer payload

FIG. 9

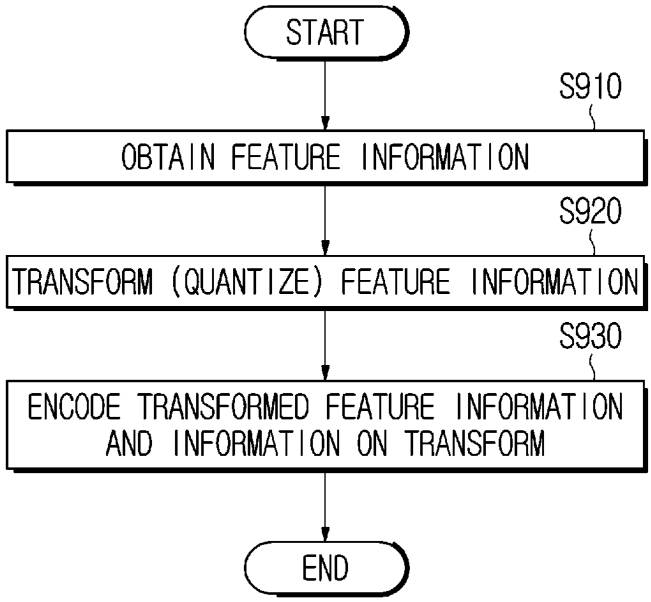

START
S910
OBTAIN FEATURE INFORMATION
S920
TRANSFORM (QUANTIZE) FEATURE INFORMATION
S930
ENCODE TRANSFORMED FEATURE INFORMATION AND INFORMATION ON TRANSFORM
END

FIG. 10

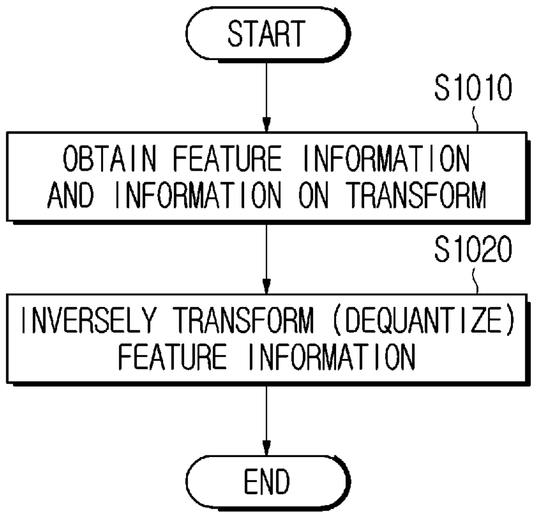

START
S1010
OBTAIN FEATURE INFORMATION AND INFORMATION ON TRANSFORM
S1020
INVERSELY TRANSFORM (DEQUANTIZE) FEATURE INFORMATION
END

FIG. 11

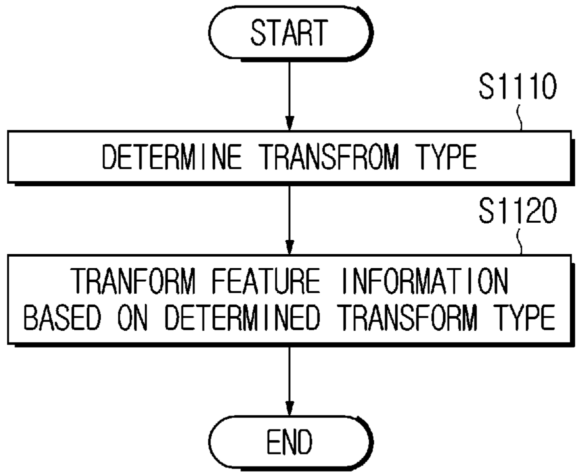

START
S1110
DETERMINE TRANSFROM TYPE
S1120
TRANFORM FEATURE INFORMATION BASED ON DETERMINED TRANSFORM TYPE
END

FIG. 12

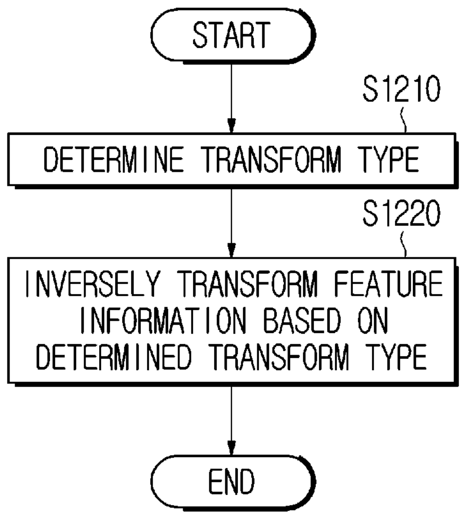

START
S1210
DETERMINE TRANSFORM TYPE
S1220
INVERSELY TRANSFORM FEATURE INFORMATION BASED ON DETERMINED TRANSFORM TYPE
END

FIG. 13

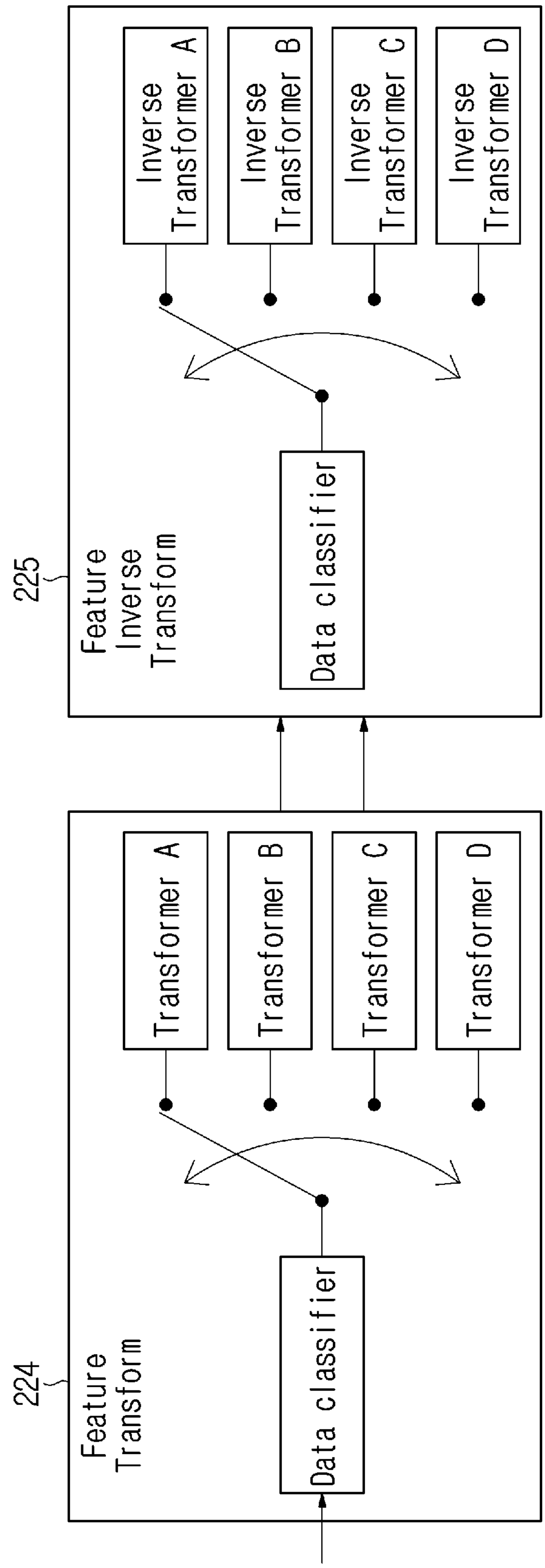

224
Feature Transform
Data classifier
Transformer A
Transformer B
Transformer C
Transformer D
225
Feature Inverse Transform
Data classifier
Inverse Transformer A
Inverse Transformer B
Inverse Transformer C
Inverse Transformer D

FIG. 14

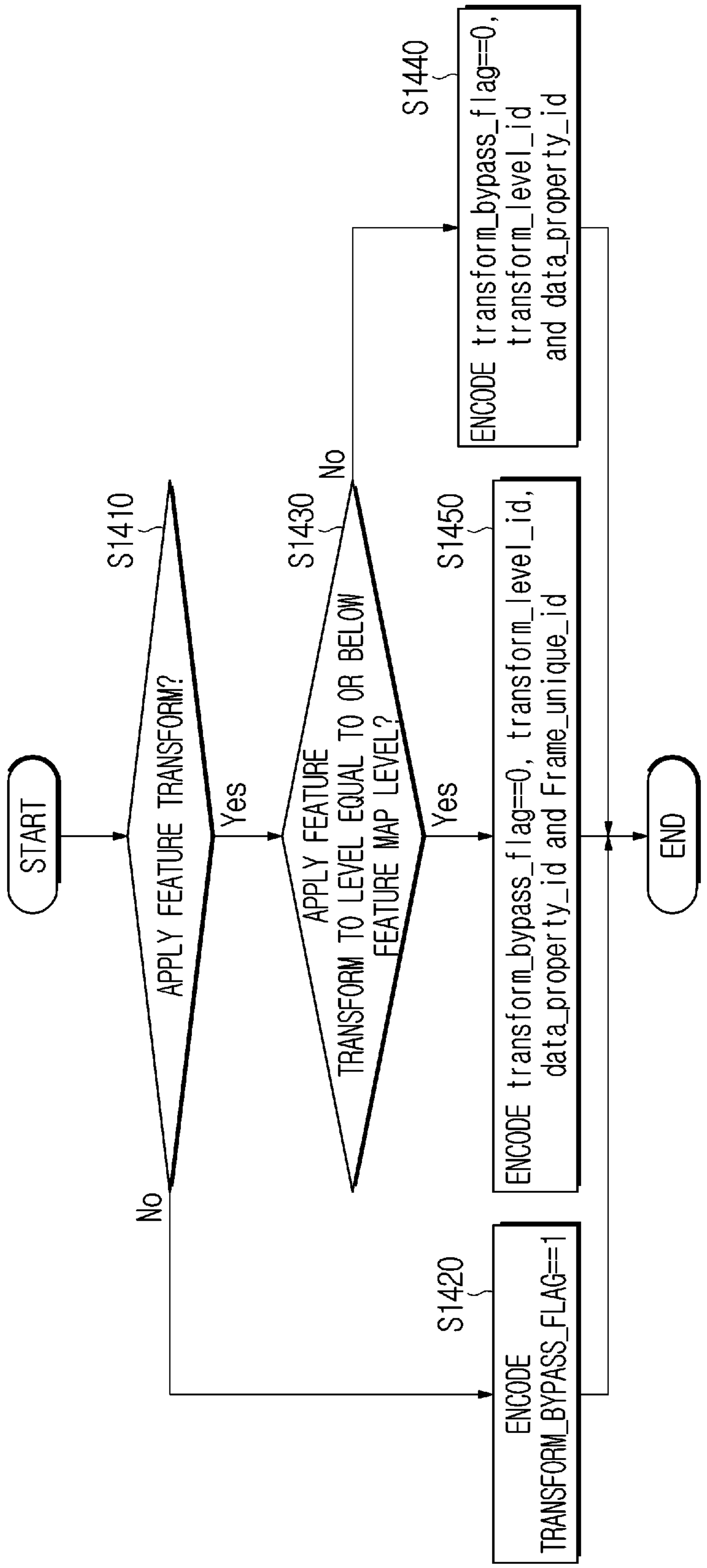

START
APPLY FEATURE TRANSFORM?
S1410
Yes
No
ENCODE TRANSFORM_BYPASS_FLAG==1
S1420
APPLY FEATURE TRANSFORM TO LEVEL EQUAL TO OR BELOW FEATURE MAP LEVEL?
S1430
Yes
No
ENCODE transform_bypass_flag==0, transform_level_id and data_property_id
S1440
ENCODE transform_bypass_flag==0, transform_level_id, data_property_id and Frame_unique_id
S1450
END

FIG. 15

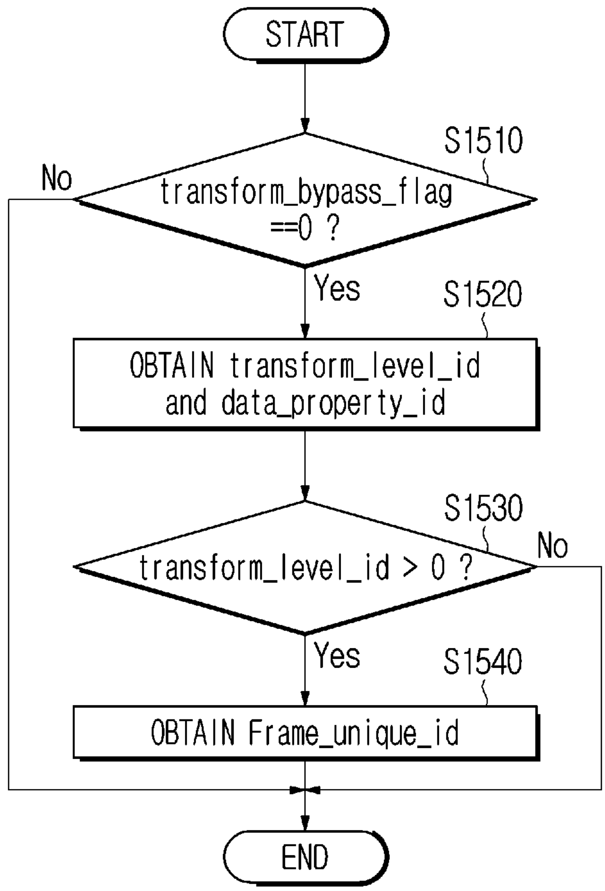

START
S1510
No
transform_bypass_flag
==0 ?
Yes
S1520
OBTAIN transform_level_id
and data_property_id
S1530
No
transform_level_id > 0 ?
Yes
S1540
OBTAIN Frame_unique_id
END

FIG. 16

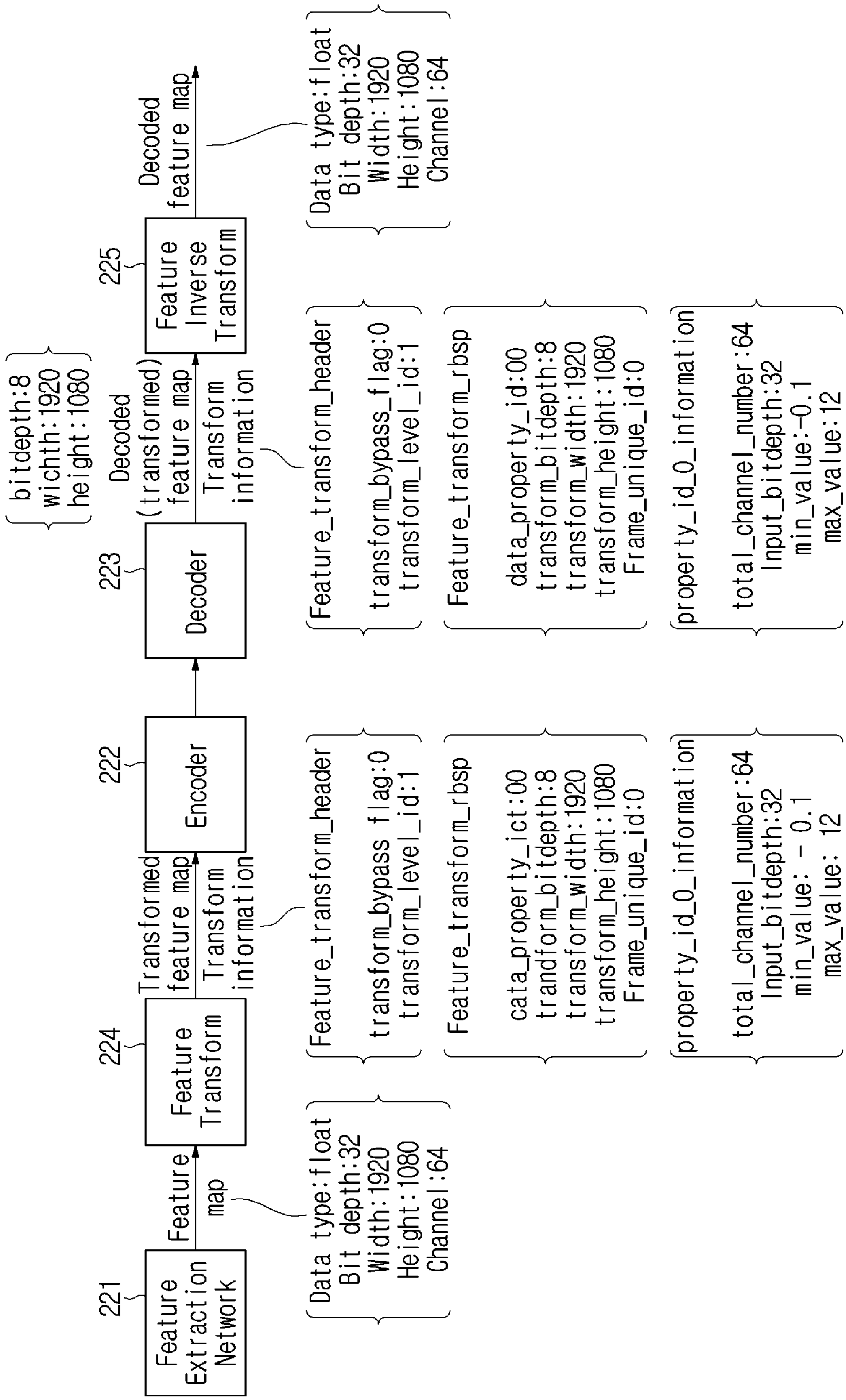

221
Feature Extraction Network
Feature map
Data type:float
Bit depth:32
Width:1920
Height:1080
Channel:64
224
Feature Transform
Transformed feature map
Transform information
Feature_transform_header
transform_bypass flag:0
transform_level_id:1
Feature_transform_rbsp
cata_property_ict:00
trandform_bitdepth:8
transform_width:1920
transform_height:1080
Frame_unique_id:0
property_id_0_information
total_channel_number:64
Input_bitdepth:32
min_value: - 0.1
max_value: 12
222
Encoder
223
Decoder
bitdepth:8
wichth:1920
height:1080
Decoded (transformed) feature map
Transform information
Feature_transform_header
transform_bypass_flag:0
transform_level_id:1
Feature_transform_rbsp
data_property_id:00
transform_bitdepth:8
transform_width:1920
transform_height:1080
Frame_unique_id:0
property_id_0_information
total_channel_number:64
Input_bitdepth:32
min_value:-0.1
max_value:12
225
Feature Inverse Transform
Decoded feature map
Data type:float
Bit depth:32
Width:1920
Height:1080
Channel:64

FIG. 17

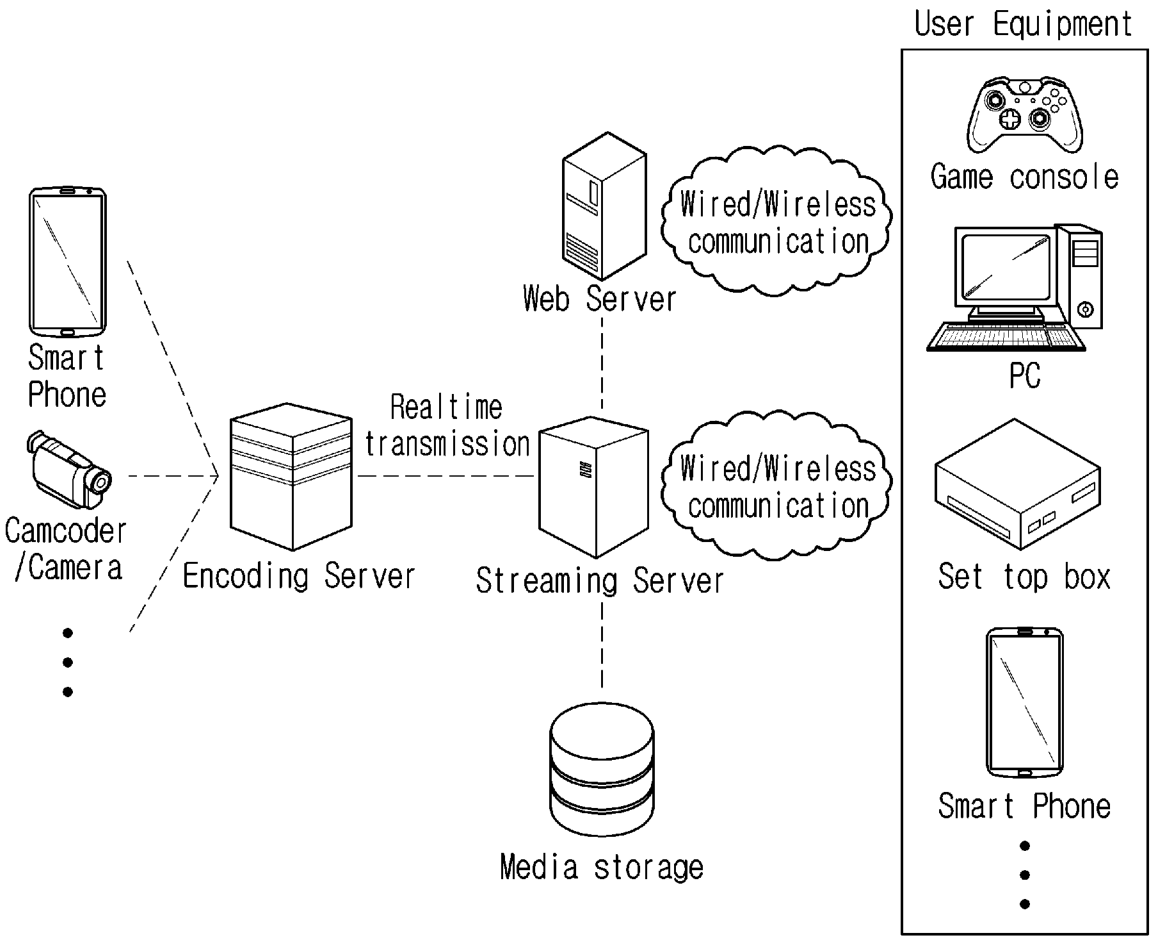
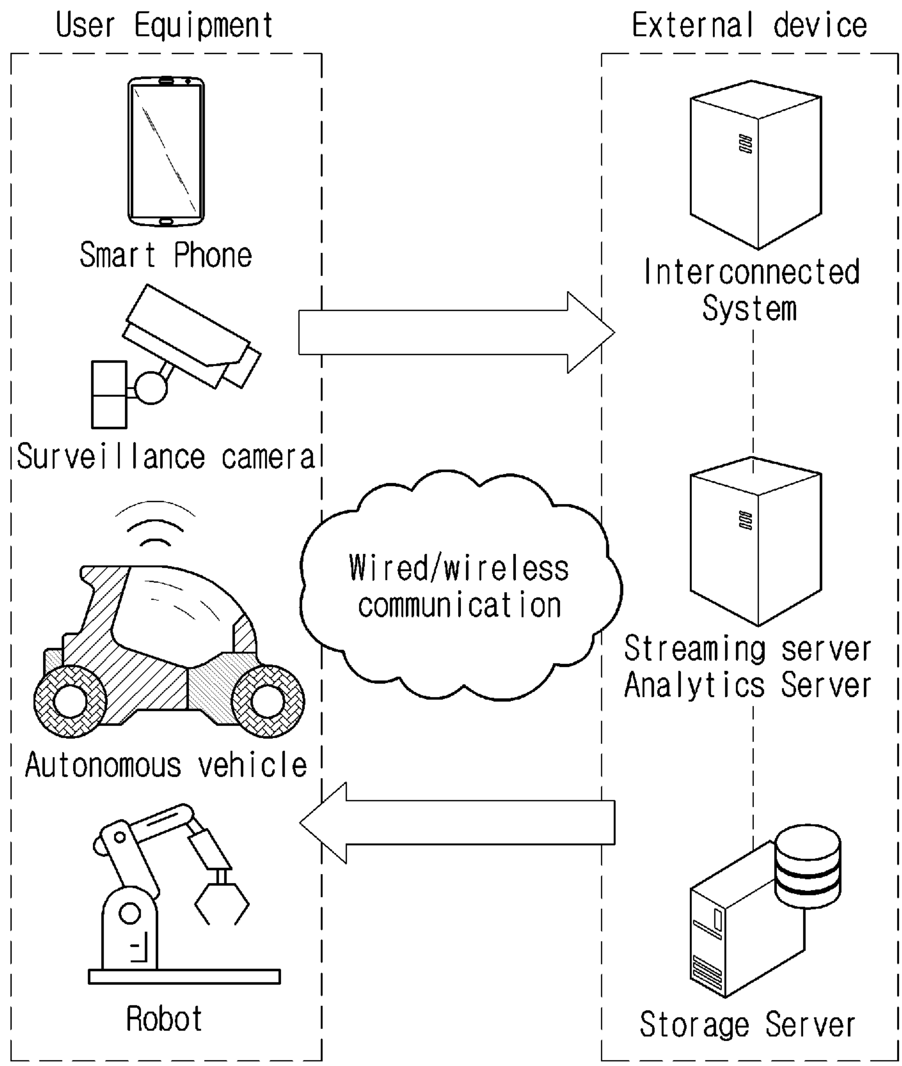

User Equipment
Game console
PC
Set top box
Smart Phone
Web Server
Wired/Wireless communication
Wired/Wireless communication
Smart Phone
Camcoder /Camera
Encoding Server
Realtime transmission
Streaming Server
Media storage

FIG. 18


User Equipment
External device
Smart Phone
Surveillance camera
Autonomous vehicle
Robot
Wired/wireless communication
Interconnected System
Streaming server
Analytics Server
Storage Server

INTERNATIONAL SEARCH REPORT

International application No.
PCT/KR2023/003070

A. CLASSIFICATION OF SUBJECT MATTER

H04N 19/136(2014.01)i; **H04N 19/70**(2014.01)i; **H04N 19/60**(2014.01)i; **H04N 19/44**(2014.01)i; **H04N 19/119**(2014.01)i; **H04N 19/186**(2014.01)i; **H04N 19/96**(2014.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04N 19/136(2014.01); G06K 9/46(2006.01); G06K 9/62(2006.01); G06N 3/04(2006.01); G06N 3/08(2006.01); H04N 19/182(2014.01); H04N 19/513(2014.01)

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Korean utility models and applications for utility models: IPC as above
Japanese utility models and applications for utility models: IPC as above

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKOMPASS (KIPO internal) & keywords: 피처(feature), 변환(transform), 역변환(inverse transform), 타입(type), 후보(candidate)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	WO 2021-177652 A1 (LG ELECTRONICS INC.) 10 September 2021 (2021-09-10) See paragraphs [0128], [0176], [0191]-[0193] and [0327]; claim 1; and figures 8, 10 and 22-25.	1-10
Y	US 2018-0249173 A1 (SHARP KABUSHIKI KAISHA) 30 August 2018 (2018-08-30) See paragraphs [0028] and [0032].	1-10
A	WO 2021-205065 A1 (NOKIA TECHNOLOGIES OY) 14 October 2021 (2021-10-14) See paragraphs [0003]-[0012]; and claims 1-11.	1-10
A	WO 2020-055279 A1 (HUAWEI TECHNOLOGIES CO., LTD. et al.) 19 March 2020 (2020-03-19) See page 2, line 13 - page 5, line 2; and claims 1-6.	1-10
A	WO 2021-205067 A1 (NOKIA TECHNOLOGIES OY) 14 October 2021 (2021-10-14) See paragraphs [0003]-[0010]; and claims 1-17.	1-10

☐ Further documents are listed in the continuation of Box C.   ☑ See patent family annex.

* Special categories of cited documents:
- "A" document defining the general state of the art which is not considered to be of particular relevance
- "D" document cited by the applicant in the international application
- "E" earlier application or patent but published on or after the international filing date
- "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- "O" document referring to an oral disclosure, use, exhibition or other means
- "P" document published prior to the international filing date but later than the priority date claimed
- "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
- "&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
26 May 2023	**30 May 2023**

Name and mailing address of the ISA/KR	Authorized officer
Korean Intellectual Property Office **Government Complex-Daejeon Building 4, 189 Cheongsa-ro, Seo-gu, Daejeon 35208**	
Facsimile No. **+82-42-481-8578**	Telephone No.

Form PCT/ISA/210 (second sheet) (July 2022)

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/KR2023/003070

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
WO 2021-177652 A1	10 September 2021	KR 10-2022-0145405 A	28 October 2022
		US 2023-0082561 A1	16 March 2023
US 2018-0249173 A1	30 August 2018	BR PI1014257 A2	12 April 2016
		CN 102388614 A	21 March 2012
		CN 104717502 A	17 June 2015
		CN 104717505 B	11 January 2019
		EA 201101473 A1	30 April 2012
		EA 201690320 A2	30 September 2016
		EA 201690324 A1	31 October 2016
		EA 201690326 A1	31 October 2016
		EA 201690329 A1	30 November 2016
		EP 2418855 A1	15 February 2012
		EP 2418855 B1	27 December 2017
		HK 1211398 A1	20 May 2016
		JP 2015-173505 A	01 October 2015
		JP 2017-028726 A	02 February 2017
		JP 6239713 B2	29 November 2017
		US 10645410 B2	05 May 2020
		US 2012-0033731 A1	09 February 2012
		US 2014-0348235 A1	27 November 2014
		US 2016-0241886 A1	18 August 2016
		US 2017-0251221 A1	31 August 2017
		WO 2010-116869 A1	14 October 2010
WO 2021-205065 A1	14 October 2021	None	
WO 2020-055279 A1	19 March 2020	CN 112673625 A	16 April 2021
		EP 3834409 A1	16 June 2021
		US 2021-0203997 A1	01 July 2021
WO 2021-205067 A1	14 October 2021	EP 4133740 A1	15 February 2023
		US 11375204 B2	28 June 2022
		US 2021-0314573 A1	07 October 2021

Form PCT/ISA/210 (patent family annex) (July 2022)